

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-358707

(43)Date of publication of application : 26.12.2001

(51)Int.Cl.

H04L 9/10

G06F 12/00

G06F 12/14

G06F 13/00

G06F 15/00

H04L 9/08

(21)Application number : 2000-179693

(71)Applicant : SONY CORP

(22)Date of filing : 15.06.2000

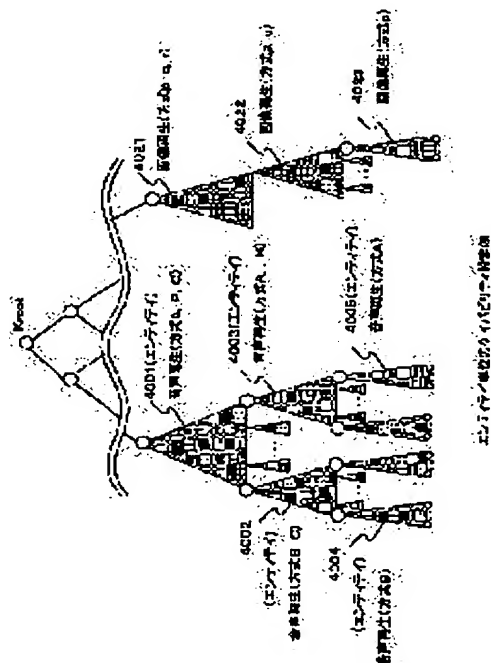
(72)Inventor : KITATANI YOSHIMICHI
ISHIGURO RYUJI
OSAWA YOSHITOMO
ASANO TOMOYUKI

(54) INFORMATION PROCESSING SYSTEM AND METHOD USING CRYPTOGRAPHIC KEY BLOCK AND PROGRAM PROVIDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To realize an information processing system and method capable of generating and distributing an effective key block(EKB) in accordance with the data throughput of a device.

SOLUTION: In an entity which sets a subtree classified on the basis of capability as the data throughput of the device in a key tree obtained by making each key correspond to routes, nodes and leaves on paths from the root of the tree constructed with a plurality of devices as leaves, a sub effective key block(sub EKB) being effective in the entity is generated. In a key issuing center, the effective key block(EKB) that can be decoded only in an entity having common capability is generated on the basis of the capability information of the entity.



LEGAL STATUS

[Date of request for examination]

26.11.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

Best Available Copy

This Page Blank (uspto)

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

This Page Blank (uspto)

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2001-358707
(P2001-358707A)

(43)公開日 平成13年12月26日(2001. 12. 26)

| (51)Int.Cl. ⁷ | 識別記号 | F I | テ-リ-ト(参考) |
|--------------------------|-------|---------------|-------------------|
| H 0 4 L 9/10 | | G 0 6 F 12/00 | 5 3 7 H 5 B 0 1 7 |
| G 0 6 F 12/00 | 5 3 7 | 12/14 | 3 2 0 B 5 B 0 8 2 |
| | 12/14 | 13/00 | 3 5 1 Z 5 B 0 8 5 |
| | 13/00 | 15/00 | 3 3 0 Z 5 B 0 8 9 |
| | 15/00 | H 0 4 L 9/00 | 6 2 1 A 5 J 1 0 4 |

審査請求 未請求 請求項の数21 O L (全 50 頁) 最終頁に続く

(21)出願番号 特願2000-179693(P2000-179693)

(22)出願日 平成12年6月15日(2000. 6. 15)

(71)出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72)発明者 北谷 義道

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72)発明者 石黒 隆二

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(74)代理人 100101801

弁理士 山田 英治 (外2名)

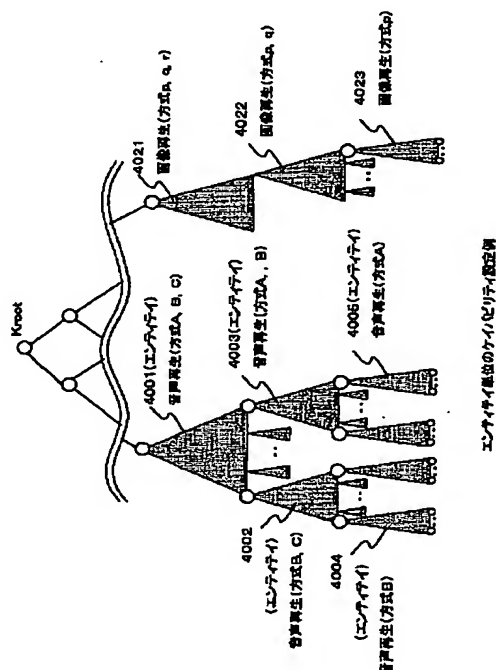
最終頁に続く

(54)【発明の名称】 暗号鍵ブロックを用いた情報処理システムおよび情報処理方法、並びにプログラム提供媒体

(57)【要約】

【課題】 デバイスのデータ処理能力に応じて有効化キーブロック (E K B) を生成して配信することを可能とした情報処理システムおよび方法を実現する。

【解決手段】 複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーに、デバイスのデータ処理能力としてのケイパビリティに基づいて区分したサブツリーを設定し、それぞれのサブツリーの管理主体であるエンティティにおいて、エンティティ内で有効なサブ有効化キーブロック (サブE K B) を生成し、キー発行センターにおいて、エンティティのケイパビリティ情報に基づいて、共通ケイパビリティを持つエンティティにおいてのみ復号可能な有効化キーブロック (E K B) を生成する構成とした。



【特許請求の範囲】

【請求項 1】複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上のキー更新、および下位キーによる上位キーの暗号化処理を実行して特定デバイスにおいてのみ復号可能な有効化キーブロック（EKB）を生成してデバイスに提供する暗号鍵ブロックを用いた情報処理システムにおいて、

前記キーツリーの一部を構成し、デバイスのデータ処理能力としてのケイパビリティに基づいて区分されたサブツリーを管理し、該サブツリーに属するノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブ EKB）を生成する複数のエンティティと、

前記複数のエンティティのケイパビリティ情報を管理し、共通のケイパビリティを持つエンティティの生成するサブ有効化キーブロック（サブ EKB）を用いて、共通のケイパビリティを持つエンティティにおいてのみ復号可能な有効化キーブロック（EKB）を生成するキー発行センター（KDC）と、
を有することを特徴とする暗号鍵ブロックを用いた情報処理システム。

【請求項 2】前記キー発行センター（KDC）は、複数のエンティティ各々の識別子と、エンティティ各々のケイパビリティ情報と、エンティティ各々のサブ有効化キーブロック（サブ EKB）情報とを対応付けたケイパビリティ管理テーブルを有し、該ケイパビリティ管理テーブルに基づいて、デバイスに対する配信データの処理可能なエンティティを選択して、該選択エンティティ配下のデバイスでのみ復号可能な有効化キーブロック（EKB）を生成する構成を有することを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 3】前記キーツリーに対する新規追加エンティティは、
該新規エンティティ内のサブツリー内のノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブ EKB）を生成し、前記キー発行センター（KDC）に対するサブ EKB の登録処理を実行するとともに、自己のエンティティのケイパビリティ情報の通知処理を実行する構成であることを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 4】前記複数のエンティティは、
1 つのエンティティの最下段の末端ノードを他のエンティティの頂点ノード（サブルート）として構成した上位エンティティおよび下位エンティティの階層化構造を有することを特徴とする請求項 1 に記載の暗号鍵ブロック

を用いた情報処理システム。

【請求項 5】前記複数のエンティティの各々は、
自己のエンティティに属するサブツリーを構成するノードまたはリーフに対応するキーの設定、更新処理権限を有する構成であることを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 6】前記複数のエンティティ中、エンティティ内の最下段リーフを個々のデバイスに対応するリーフとした最下層のエンティティに属するデバイスの各々は、
10 自己の属するエンティティの頂点ノード（サブルート）から自己のデバイスに対応するリーフに至るパス上のノード、リーフに設定されたノードキーおよびリーフキーを格納した構成を有することを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 7】前記複数のエンティティの各々は、
自己のエンティティの下位に、さらに自己管理エンティティを追加するため、自己のエンティティ内の最下段のノードまたはリーフ中の 1 以上のノードまたはリーフをリザーブノードとして保留して設定した構成を有することを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 8】新規エンティティを末端ノードに追加する上位エンティティは、
新規エンティティのサブツリーを設定するノードである上位エンティティ末端ノードに対応するキーを、前記新規エンティティの頂点ノード（サブルート）キーとして設定する構成であることを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 9】デバイスのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・デバイスに対応するリーフに至るパス上のノードに設定されたノードキーを更新し、更新ノードキーをリボークデバイス以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成した更新サブ EKB を生成して上位エンティティに送信し、上位エンティティは更新サブ EKB を提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブ EKB を生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブ EKB 生成および送信処理を順次実行して、リボークデバイスからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブ EKB の前記キー発行センター（KDC）への登録処理を行なうことにより、デバイスのリボーク処理を実行する構成を有することを特徴とする請求項 1 に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項 10】下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上のノードに設定されたノードキーを

更新した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター(KDC)への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行する構成を有することを特徴とする請求項1に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項11】下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード(サブルート)からリボーク・エンティティに対応する末端ノードに至るパス上の、該末端ノードを除くノードに設定されたノードキーを更新した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のリボーク・エンティティに対応する末端ノードを除くノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター(KDC)への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行する構成を有することを特徴とする請求項1に記載の暗号鍵ブロックを用いた情報処理システム。

【請求項12】複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上のキー更新、および下位キーによる上位キーの暗号化処理を実行して特定デバイスにおいてのみ復号可能な有効化キーブロック(EKB)を生成してデバイスに提供する情報処理システムにおける暗号鍵ブロックを用いた情報処理方法において、

前記キーツリーの一部を構成し、デバイスのデータ処理能力としてのケイパビリティに基づいて区分されたサブツリーを管理するエンティティにおいて、各エンティティのサブツリーに属するノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック(サブEKB)を生成するステップと、

前記複数のエンティティのケイパビリティ情報を保有するキー発行センター(KDC)において、前記複数のエンティティのケイパビリティ情報に基づいて、共通のケイパビリティを持つエンティティの生成するサブ有効化

キーブロック(サブEKB)を抽出し共通のケイパビリティを持つエンティティにおいてのみ復号可能な有効化キーブロック(EKB)を生成するステップと、を有することを特徴とする暗号鍵ブロックを用いた情報処理方法。

【請求項13】前記キー発行センター(KDC)における有効化キーブロック(EKB)生成ステップは、共通のケイパビリティを持つエンティティを選択するエンティティ選択ステップと、

10 前記エンティティ選択ステップにおいて選択されたエンティティによって構成されるエンティティ・ツリーを生成するステップと、

前記エンティティ・ツリーを構成するノードキーを更新するノードキー更新ステップと、

前記ノードキー更新ステップにおいて更新したノードキー、および選択エンティティのサブEKBに基づいて選択エンティティにおいてのみ復号可能な有効化キーブロック(EKB)を生成するステップと、を含むことを特徴とする請求項12に記載の暗号鍵ブロックを用いた情報処理方法。

20 【請求項14】前記情報処理方法において、さらに、前記キー発行センター(KDC)は、複数のエンティティ各々の識別子と、エンティティ各々のケイパビリティ情報と、エンティティ各々のサブ有効化キーブロック(サブEKB)情報とを対応付けたケイパビリティ管理テーブルを有し、該ケイパビリティ管理テーブルに基づいて、デバイスに対する配信データの処理可能なエンティティを選択して、該選択エンティティ配下のデバイスでのみ復号可能な有効化キーブロック

30 (EKB)を生成することを特徴とする請求項12に記載の暗号鍵ブロックを用いた情報処理方法。

【請求項15】前記情報処理方法において、さらに、前記キーツリーに対する新規追加エンティティは、該新規エンティティ内のサブツリー内のノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック(サブEKB)を生成し、前記キー発行センター(KDC)に対するサブEKBの登録処理を実行するとともに、自己のエンティティのケイパビリティ情報の通知処理を実行することを特徴とする請求項12に記載の暗号鍵ブロックを用いた情報処理方法。

40 【請求項16】前記情報処理方法において、さらに、前記複数のエンティティの各々は、自己のエンティティに属するサブツリーを構成するノードまたはリーフに対応するキーの設定、更新処理を実行することを特徴とする請求項12に記載の暗号鍵ブロックを用いた情報処理方法。

【請求項17】前記情報処理方法において、さらに、新規エンティティを末端ノードに追加する上位エンティティは、

50 新規エンティティのサブツリーを設定するノードである

上位エンティティ末端ノードに対応するキーを、前記新規エンティティの頂点ノード（サブルート）キーとして設定することを特徴とする請求項 12 に記載の暗号鍵ブロックを用いた情報処理方法。

【請求項 18】 デバイスのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・デバイスに対応するリーフに至るパス上のノードに設定されたノードキーを更新し、更新ノードキーをリボークデバイス以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成した更新サブ E K B を生成して上位エンティティに送信し、上位エンティティは更新サブ E K B を提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブ E K B を生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブ E K B 生成および送信処理を順次実行して、リボークデバイスからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブ E K B の前記キー発行センター（K D C）への登録処理を行なうことにより、デバイスのリボーク処理を実行することを特徴とする請求項 12 に記載の暗号鍵ブロックを用いた情報処理方法。

【請求項 19】 下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上のノードに設定されたノードキーを更新した更新サブ E K B を生成して上位エンティティに送信し、上位エンティティは更新サブ E K B を提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブ E K B を生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブ E K B 生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブ E K B の前記キー発行センター（K D C）への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行することを特徴とする請求項 12 に記載の暗号鍵ブロックを用いた情報処理方法。

【請求項 20】 下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上の、該末端ノードを除くノードに設定されたノードキーを更新した更新サブ E K B を生成して上位エンティティに送信し、上位エンティティは更新サブ E K B を提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブ E K B を生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブ E K B 生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のリボーク・エンティティ

に対応する末端ノードを除くノードキー更新を行ない、キー更新により生成された更新サブ E K B の前記キー発行センター（K D C）への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行することを特徴とする請求項 12 に記載の暗号鍵ブロックを用いた情報処理方法。

【請求項 21】 複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上のキー更新、および下位キーによる上位キーの暗号化処理を実行して特定デバイスにおいてのみ復号可能な有効化キープロック（E K B）を生成してデバイスに提供する情報処理システムにおける有効化キープロック（E K B）生成処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、前記キーツリーの一部を構成し、デバイスのデータ処理能力としてのケイパビリティに基づいて区分されたサブツリーを管理するエンティティにおいて、各エンティティのサブツリーに属するノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キープロック（サブ E K B）を生成するステップと、前記複数のエンティティのケイパビリティ情報を保有するキー発行センター（K D C）において、前記複数のエンティティのケイパビリティ情報に基づいて、共通のケイパビリティを持つエンティティの生成するサブ有効化キープロック（サブ E K B）を抽出し共通のケイパビリティを持つエンティティにおいてのみ復号可能な有効化キープロック（E K B）を生成するステップと、を含むことを特徴とするプログラム提供媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、暗号鍵ブロックを用いた情報処理システムおよび情報処理方法、並びにプログラム提供媒体に関し、特に、暗号処理を伴うシステムにおける暗号処理鍵を配信するシステムおよび方法に関する。特に、木構造の階層的鍵配信方式を用いることにより、メッセージ量を小さく抑えて、例えばコンテンツキー配信、あるいは各種鍵の更新の際のデータ配信の負荷を軽減し、かつデータの安全性を保持することを可能とするとともに、階層的鍵配信ツリーを管理下のデバイスのデータ処理能力としてのケイパビリティに基づいて区分したサブツリーとしてのエンティティで管理する構成としてケイパビリティに基づく鍵配信および管理構成を実現した暗号鍵ブロックを用いた情報処理システムおよび情報処理方法、並びにプログラム提供媒体に関する。

【0002】

【従来の技術】昨今、ゲームプログラム、音声データ、画像データ等、様々なソフトウェアデータ（以下、これらをコンテンツ（Content）と呼ぶ）を、インターネット等のネットワーク、あるいはDVD、CD等の流通可能な記憶媒体を介しての流通が盛んになってきている。これらの流通コンテンツは、ユーザの所有するPC（Personal Computer）、ゲーム機器によってデータ受信、あるいは記憶媒体の装着がなされて再生されたり、あるいはPC等のに付属する記録再生機器内の記録デバイス、例えばメモリカード、ハードディスク等に格納されて、格納媒体からの新たな再生により利用される。

【0003】ビデオゲーム機器、PC等の情報機器には、流通コンテンツをネットワークから受信するため、あるいはDVD、CD等にアクセスするためのインタフェースを有し、さらにコンテンツの再生に必要な制御手段、プログラム、データのメモリ領域として使用されるRAM、ROM等を有する。

【0004】音楽データ、画像データ、あるいはプログラム等の様々なコンテンツは、再生機器として利用されるゲーム機器、PC等の情報機器本体からのユーザ指示、あるいは接続された入力手段を介したユーザの指示により記憶媒体から呼び出され、情報機器本体、あるいは接続されたディスプレイ、スピーカ等を通じて再生される。

【0005】ゲームプログラム、音楽データ、画像データ等、多くのソフトウェア・コンテンツは、一般的にその作成者、販売者に頒布権等が保有されている。従って、これらのコンテンツの配布に際しては、一定の利用制限、すなわち、正規ユーザに対してのみ、ソフトウェアの使用を許諾し、許可のない複製等が行われないようにする、すなわちセキュリティを考慮した構成をとるのが一般的となっている。

【0006】ユーザに対する利用制限を実現する1つの手法が、配布コンテンツの暗号化処理である。すなわち、例えばインターネット等を介して暗号化された音声データ、画像データ、ゲームプログラム等の各種コンテンツを配布するとともに、正規ユーザであると確認された者に対してのみ、配布された暗号化コンテンツを復号する手段、すなわち復号鍵を付与する構成である。

【0007】暗号化データは、所定の手続きによる復号化処理によって利用可能な復号データ（平文）に戻すことができる。このような情報の暗号化処理に暗号化鍵を用い、復号化処理に復号化鍵を用いるデータ暗号化、復号化方法は従来からよく知られている。

【0008】暗号化鍵と復号化鍵を用いるデータ暗号化・復号化方法の態様には様々な種類があるが、その1つの例としていわゆる共通鍵暗号化方式と呼ばれている方式がある。共通鍵暗号化方式は、データの暗号化処理に用いる暗号化鍵とデータの復号化に用いる復号化鍵を共通のものとして、正規のユーザにこれら暗号化処理、復号

化に用いる共通鍵を付与して、鍵を持たない不正ユーザによるデータアクセスを排除するものである。この方式の代表的な方式にDES（データ暗号標準：Data encryption standard）がある。

【0009】上述の暗号化処理、復号化に用いられる暗号化鍵、復号化鍵は、例えばあるパスワード等に基づいてハッシュ関数等の一方向性関数を適用して得ることができる。一方向性関数とは、その出力から逆に入力を求めるのは非常に困難となる関数である。例えばユーザが決めたパスワードを入力として一方向性関数を適用して、その出力に基づいて暗号化鍵、復号化鍵を生成するものである。このようにして得られた暗号化鍵、復号化鍵から、逆にそのオリジナルのデータであるパスワードを求めることは実質上不可能となる。

【0010】また、暗号化するとき使用する暗号化鍵による処理と、復号するとき使用する復号化鍵の処理とを異なるアルゴリズムとした方式がいわゆる公開鍵暗号化方式と呼ばれる方式である。公開鍵暗号化方式は、不特定のユーザが使用可能な公開鍵を使用する方法であり、特定個人に対する暗号化文書を、その特定個人が発行した公開鍵を用いて暗号化処理を行なう。公開鍵によって暗号化された文書は、その暗号化処理に使用された公開鍵に対応する秘密鍵によってのみ復号処理が可能となる。秘密鍵は、公開鍵を発行した個人のみが所有するので、その公開鍵によって暗号化された文書は秘密鍵を持つ個人のみが復号することができる。公開鍵暗号化方式の代表的なものにはRSA（Rivest-Shamir-Adleman）暗号がある。このような暗号化方式を利用することにより、暗号化コンテンツを正規ユーザに対してのみ復号可能とするシステムが可能となる。

【0011】

【発明が解決しようとする課題】上記のようなコンテンツ配信システムでは、コンテンツを暗号化してユーザにネットワーク、あるいはDVD、CD等の記録媒体に格納して提供し、暗号化コンテンツを復号するコンテンツキーを正当なユーザにのみ提供する構成が多く採用されている。コンテンツキー自体の不正なコピー等を防ぐためのコンテンツキーを暗号化して正当なユーザに提供し、正当なユーザのみが有する復号キーを用いて暗号化コンテンツキーを復号してコンテンツキーを使用可能とする構成が提案されている。

【0012】正当なユーザであるか否かの判定は、一般には、例えばコンテンツの送信者であるコンテンツプロバイダとユーザデバイス間において、コンテンツ、あるいはコンテンツキーの配信前に認証処理を実行することによって行なう。一般的な認証処理においては、相手の確認を行なうとともに、その通信でのみ有効なセッションキーを生成して、認証が成立した場合に、生成したセッションキーを用いてデータ、例えばコンテンツあるいはコンテンツキーを暗号化して通信を行なう。認証方式

には、共通鍵暗号方式を用いた相互認証と、公開鍵方式を使用した認証方式があるが、共通鍵を使った認証においては、システムワイドで共通な鍵が必要になり、更新処理等の際に不便である。また、公開鍵方式においては、計算負荷が大きくまた必要なメモリ量も大きくなり、各デバイスにこのような処理手段を設けることは望ましい構成とはいえない。

【0013】本発明では、上述のようなデータの送信者、受信者間の相互認証処理に頼ることなく、正当なユーザに対してのみ、安全にデータを送信することを可能とするとともに、階層的鍵配信ツリーを管理下のデバイスのデータ処理能力としての鍵パブリシティに基づいて区分したサブツリーとしてのエンティティで管理する構成として鍵パブリシティに基づく鍵配信および管理構成を実現した暗号鍵ブロックを用いた情報処理システムおよび情報処理方法、並びにプログラム提供媒体を提供することを目的とする。

【0014】

【課題を解決するための手段】本発明の第1の側面は、複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上のキー更新、および下位キーによる上位キーの暗号化処理を実行して特定デバイスにおいてのみ復号可能な有効化キーブロック（EKB）を生成してデバイスに提供する暗号鍵ブロックを用いた情報処理システムにおいて、前記キーツリーの一部を構成し、デバイスのデータ処理能力としての鍵パブリシティに基づいて区分されたサブツリーを管理し、該サブツリーに属するノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブEKB）を生成する複数のエンティティと、前記複数のエンティティの鍵パブリシティ情報を管理し、共通の鍵パブリシティを持つエンティティの生成するサブ有効化キーブロック（サブEKB）を用いて、共通の鍵パブリシティを持つエンティティにおいてのみ復号可能な有効化キーブロック（EKB）を生成するキー発行センター（KDC）と、を有することを特徴とする暗号鍵ブロックを用いた情報処理システムにある。

【0015】さらに、本発明の情報処理システムにおいて、前記キー発行センター（KDC）は、複数のエンティティ各々の識別子と、エンティティ各々の鍵パブリシティ情報と、エンティティ各々のサブ有効化キーブロック（サブEKB）情報とを対応付けた鍵パブリシティ管理テーブルを有し、該鍵パブリシティ管理テーブルに基づいて、デバイスに対する配信データの処理可能なエンティティを選択して、該選択エンティティ配下のデバイスでのみ復号可能な有効化キーブロック（EKB）を生成する構成を有することを特徴とする。

【0016】さらに、本発明の情報処理システムにおい

て、前記キーツリーに対する新規追加エンティティは、該新規エンティティ内のサブツリー内のノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブEKB）を生成し、前記キー発行センター（KDC）に対するサブEKBの登録処理を実行するとともに、自己のエンティティの鍵パブリシティ情報の通知処理を実行する構成であることを特徴とする。

【0017】さらに、本発明の情報処理システムにおいて、前記複数のエンティティは、1つのエンティティの最下段の末端ノードを他のエンティティの頂点ノード（サブルート）として構成した上位エンティティおよび下位エンティティの階層化構造を有することを特徴とする。

【0018】さらに、本発明の情報処理システムにおいて、前記複数のエンティティの各々は、自己のエンティティに属するサブツリーを構成するノードまたはリーフに対応するキーの設定、更新処理権限を有する構成であることを特徴とする。

【0019】さらに、本発明の情報処理システムにおいて、前記複数のエンティティ中、エンティティ内の最下段リーフを個々のデバイスに対応するリーフとした最下層のエンティティに属するデバイスの各々は、自己の属するエンティティの頂点ノード（サブルート）から自己のデバイスに対応するリーフに至るパス上のノード、リーフに設定されたノードキーおよびリーフキーを格納した構成を有することを特徴とする。

【0020】さらに、本発明の情報処理システムにおいて、前記複数のエンティティの各々は、自己のエンティティの下位に、さらに自己管理エンティティを追加するため、自己のエンティティ内の最下段のノードまたはリーフ中の1以上のノードまたはリーフをリザーブノードとして保留して設定した構成を有することを特徴とする。

【0021】さらに、本発明の情報処理システムにおいて、新規エンティティを末端ノードに追加する上位エンティティは、新規エンティティのサブツリーを設定するノードである上位エンティティ末端ノードに対応するキーを、前記新規エンティティの頂点ノード（サブルート）キーとして設定する構成であることを特徴とする。

【0022】さらに、本発明の情報処理システムにおいて、デバイスのリポーカ処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリポーカ・デバイスに対応するリーフに至るパス上のノードに設定されたノードキーを更新し、更新ノードキーをリポーカデバイス以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位

エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボークデバイスからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター（KDC）への登録処理を行なうことにより、デバイスのリボーク処理を実行する構成を有することを特徴とする。

【0023】さらに、本発明の情報処理システムにおいて、下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上のノードに設定されたノードキーを更新した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター（KDC）への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行する構成を有することを特徴とする。

【0024】さらに、本発明の情報処理システムにおいて、下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上の、該末端ノードを除くノードに設定されたノードキーを更新した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のリボーク・エンティティに対応する末端ノードを除くノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター（KDC）への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行する構成を有することを特徴とする。

【0025】さらに、本発明の第2の側面は、複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上のキー更新、および下位キーによる上位キーの暗号化処理を実行して特定デバイスにおいてのみ復号可能な有効化キーブロック（EKB）を生成してデバイスに提供する情報処理システムにおける暗号鍵ブロックを用いた情報処理方法におい

て、前記キーツリーの一部を構成し、デバイスのデータ処理能力としてのケイパビリティに基づいて区分されたサブツリーを管理するエンティティにおいて、各エンティティのサブツリーに属するノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブEKB）を生成するステップと、前記複数のエンティティのケイパビリティ情報を保有するキー発行センター（KDC）において、前記複数のエンティティのケイパビリティ情報に基づいて、共通のケイパビリティを持つエンティティの生成するサブ有効化キーブロック（サブEKB）を抽出し共通のケイパビリティを持つエンティティにおいてのみ復号可能な有効化キーブロック（EKB）を生成するステップと、を有することを特徴とする暗号鍵ブロックを用いた情報処理方法にある。

【0026】さらに、本発明の情報処理方法において、前記キー発行センター（KDC）における有効化キーブロック（EKB）生成ステップは、共通のケイパビリティを持つエンティティを選択するエンティティ選択ステップと、前記エンティティ選択ステップにおいて選択されたエンティティによって構成されるエンティティ・ツリーを生成するステップと、前記エンティティ・ツリーを構成するノードキーを更新するノードキー更新ステップと、前記ノードキー更新ステップにおいて更新したノードキー、および選択エンティティのサブEKBに基づいて選択エンティティにおいてのみ復号可能な有効化キーブロック（EKB）を生成するステップとを含むことを特徴とする。

【0027】さらに、本発明の情報処理方法において、前記キー発行センター（KDC）は、複数のエンティティ各々の識別子と、エンティティ各々のケイパビリティ情報と、エンティティ各々のサブ有効化キーブロック（サブEKB）情報とを対応付けたケイパビリティ管理テーブルを有し、該ケイパビリティ管理テーブルに基づいて、デバイスに対する配信データの処理可能なエンティティを選択して、該選択エンティティ配下のデバイスでのみ復号可能な有効化キーブロック（EKB）を生成することを特徴とする。

【0028】さらに、本発明の情報処理方法において、前記キーツリーに対する新規追加エンティティは、該新規エンティティ内のサブツリー内のノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブEKB）を生成し、前記キー発行センター（KDC）に対するサブEKBの登録処理を実行するとともに、自己のエンティティのケイパビリティ情報の通知処理を実行することを特徴とする。

【0029】さらに、本発明の情報処理方法において、前記複数のエンティティの各々は、自己のエンティティに属するサブツリーを構成するノードまたはリーフに対応するキーの設定、更新処理を実行することを特徴とする。

【0030】さらに、本発明の情報処理方法において、前記情報処理方法において、さらに、新規エンティティを末端ノードに追加する上位エンティティは、新規エンティティのサブツリーを設定するノードである上位エンティティ末端ノードに対応するキーを、前記新規エンティティの頂点ノード（サブルート）キーとして設定することを特徴とする。

【0031】さらに、本発明の情報処理方法において、デバイスのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・デバイスに対応するリーフに至るパス上のノードに設定されたノードキーを更新し、更新ノードキーをリボークデバイス以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボークデバイスからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター（KDC）への登録処理を行なうことにより、デバイスのリボーク処理を実行することを特徴とする。

【0032】さらに、本発明の情報処理方法において、下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上のノードに設定されたノードキーを更新した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター（KDC）への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行することを特徴とする。

【0033】さらに、本発明の情報処理方法において、下位エンティティのリボーク処理を実行するエンティティは、エンティティ内の頂点ノード（サブルート）からリボーク・エンティティに対応する末端ノードに至るパス上の、該末端ノードを除くノードに設定されたノードキーを更新した更新サブEKBを生成して上位エンティティに送信し、上位エンティティは更新サブEKBを提供した末端ノードから自己のサブルートに至るパス上のノードキーを更新した更新サブEKBを生成してさらに上位エンティティに送信し、最上位エンティティまで、

エンティティ単位での更新サブEKB生成および送信処理を順次実行して、リボーク・エンティティからルートに至るパス上のリボーク・エンティティに対応する末端ノードを除くノードキー更新を行ない、キー更新により生成された更新サブEKBの前記キー発行センター（KDC）への登録処理を行なうことにより、エンティティ単位のリボーク処理を実行することを特徴とする。

【0034】さらに、本発明の第3の側面は、複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーを構成し、該キーツリーを構成するパスを選択して選択パス上のキー更新、および下位キーによる上位キーの暗号化処理を実行して特定デバイスにおいてのみ復号可能な有効化キーブロック（EKB）を生成してデバイスに提供する情報処理システムにおける有効化キーブロック（EKB）生成処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、前記キーツリーの一部を構成し、デバイスのデータ処理能力としてのケイパビリティに基づいて区分されたサブツリーを管理するエンティティにおいて、各エンティティのサブツリーに属するノードまたはリーフに対応して設定されるキーのみに基づくサブ有効化キーブロック（サブEKB）を生成するステップと、前記複数のエンティティのケイパビリティ情報を保有するキー発行センター（KDC）において、前記複数のエンティティのケイパビリティ情報に基づいて、共通のケイパビリティを持つエンティティの生成するサブ有効化キーブロック（サブEKB）を抽出し共通のケイパビリティを持つエンティティにおいてのみ復号可能な有効化キーブロック（EKB）を生成するステップと、を含むことを特徴とするプログラム提供媒体にある。

【0035】

【作用】本発明の構成においては、ツリー（木）構造の階層的構造の暗号化鍵配信構成を用いることにより、キー更新に必要な配信メッセージ量を小さく抑さえている。すなわち、各機器をn分木の各葉（リーフ）に配置した構成の鍵配信方法を用い、記録媒体もしくは通信回線を介して、例えばコンテンツデータの暗号鍵であるコンテンツキーもしくは認証処理に用いる認証キー、あるいはプログラムコード等を有効化キーブロックとともに配信する構成としている。このようにすることにより、正当なデバイスのみが復号可能なデータを安全に配信することが可能となる。

【0036】さらに、本発明の構成においては、階層的鍵配信ツリーを、管理下のデバイスのデータ処理能力としてのケイパビリティに基づいて区分したサブツリーとしてのエンティティで管理する構成としてケイパビリティに基づく鍵配信および管理構成を実現している。

【0037】なお、本発明の第3の側面に係るプログラム提供媒体は、例えば、様々なプログラム・コードを実行可能な汎用コンピュータ・システムに対して、コンピュータ・プログラムをコンピュータ可読な形式で提供する媒体である。媒体は、CDやFD、MOなどの記録媒体、あるいは、ネットワークなどの伝送媒体など、その形態は特に限定されない。

【0038】このようなプログラム提供媒体は、コンピュータ・システム上で所定のコンピュータ・プログラムの機能を実現するための、コンピュータ・プログラムと提供媒体との構造上又は機能上の協働的關係を定義したものである。換言すれば、該提供媒体を介してコンピュータ・プログラムをコンピュータ・システムにインストールすることによって、コンピュータ・システム上では協働的作用が発揮され、本発明の他の側面と同様の作用効果を得ることができるのである。

【0039】本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づくより詳細な説明によって明らかになるであろう。

【0040】

【発明の実施の形態】〔システム概要〕図1に本発明のデータ処理システムが適用可能なコンテンツ配信システム例を示す。コンテンツの配信側10は、コンテンツ受信側20の有する様々なコンテンツ再生可能な機器に対してコンテンツ、あるいはコンテンツキーを暗号化して送信する。受信側20における機器では、受信した暗号化コンテンツ、あるいは暗号化コンテンツキー等を復号してコンテンツあるいはコンテンツキーを取得して、画像データ、音声データの再生、あるいは各種プログラムの実行等を行なう。コンテンツの配信側10とコンテンツ受信側20との間のデータ交換は、インターネット等のネットワークを介して、あるいはDVD、CD等の流通可能な記憶媒体を介して実行される。

【0041】コンテンツの配信側10のデータ配信手段としては、インターネット11、衛星放送12、電話回線13、DVD、CD等のメディア14等があり、一方、コンテンツ受信側20のデバイスとしては、パーソナルコンピュータ(PC)21、ポータブルデバイス(PD)22、携帯電話、PDA(Personal Digital Assistants)等の携帯機器23、DVD、CDプレーヤ等の記録再生器24、ゲーム端末等の再生専用器25等がある。これらコンテンツ受信側20の各デバイスは、コンテンツ配信側10から提供されたコンテンツをネットワーク等の通信手段あるいは、あるいはメディア30から取得する。

【0042】〔デバイス構成〕図2に、図1に示すコンテンツ受信側20のデバイスの一例として、記録再生装置100の構成ブロック図を示す。記録再生装置100は、入出力I/F(Interface)120、MPEG(Moving Picture Experts Group)コーデック130、A/D、

D/Aコンバータ141を備えた入出力I/F(Interface)140、暗号処理手段150、ROM(Read Only Memory)160、CPU(Central Processing Unit)170、メモリ180、記録媒体195のドライブ190を有し、これらはバス110によって相互に接続されている。

【0043】入出力I/F120は、外部から供給される画像、音声、プログラム等の各種コンテンツを構成するデジタル信号を受信し、バス110上に出力するとともに、バス110上のデジタル信号を受信し、外部に出力する。MPEGコーデック130は、バス110を介して供給されるMPEG符号化されたデータを、MPEGデコードし、入出力I/F140に出力するとともに、入出力I/F140から供給されるデジタル信号をMPEGエンコードしてバス110上に出力する。入出力I/F140は、A/D、D/Aコンバータ141を内蔵している。入出力I/F140は、外部から供給されるコンテンツとしてのアナログ信号を受信し、A/D、D/Aコンバータ141でA/D(Analog Digital)変換することで、デジタル信号として、MPEGコーデック130に出力するとともに、MPEGコーデック130からのデジタル信号を、A/D、D/Aコンバータ141でD/A(Digital Analog)変換することで、アナログ信号として、外部に出力する。

【0044】暗号処理手段150は、例えば、1チップのLSI(Large Scale Integrated Circuit)で構成され、バス110を介して供給されるコンテンツとしてのデジタル信号の暗号化、復号処理、あるいは認証処理を実行し、暗号データ、復号データ等をバス110上に出力する構成を持つ。なお、暗号処理手段150は1チップLSIに限らず、各種のソフトウェアまたはハードウェアを組み合わせた構成によって実現することも可能である。ソフトウェア構成による処理手段としての構成については後段で説明する。

【0045】ROM160は、記録再生装置によって処理されるプログラムデータを格納する。CPU170は、ROM160、メモリ180に記憶されたプログラムを実行することで、MPEGコーデック130や暗号処理手段150等を制御する。メモリ180は、例えば、不揮発性メモリで、CPU170が実行するプログラムや、CPU170の動作上必要なデータ、さらにデバイスによって実行される暗号処理に使用されるキーセットを記憶する。キーセットについては後段で説明する。ドライブ190は、デジタルデータを記録再生可能な記録媒体195を駆動することにより、記録媒体195からデジタルデータを読み出し(再生し)、バス110上に出力するとともに、バス110を介して供給されるデジタルデータを、記録媒体195に供給して記録させる。

【0046】記録媒体195は、例えば、DVD、CD

等の光ディスク、光磁気ディスク、磁気ディスク、磁気テープ、あるいはRAM等の半導体メモリ等のデジタルデータの記憶可能な媒体であり、本実施の形態では、ドライブ190に対して着脱可能な構成であるとする。但し、記録媒体195は、記録再生装置100に内蔵する構成としてもよい。

【0047】なお、図2に示す暗号処理手段150は、1つのワンチップLSIとして構成してもよく、また、ソフトウェア、ハードウェアを組み合わせた構成によって実現する構成としてもよい。

【0048】[キー配信構成としてのツリー(木)構造について]次に、図1に示すコンテンツ配信側10からコンテンツ受信側20の各デバイスに暗号データを配信する場合における各デバイスにおける暗号処理鍵の保有構成およびデータ配信構成を図3を用いて説明する。

【0049】図3の最下段に示すナンバ0~15がコンテンツ受信側20の個々のデバイスである。すなわち図3に示す階層ツリー(木)構造の各葉(リーフ:leaf)がそれぞれのデバイスに相当する。

【0050】各デバイス0~15は、製造時あるいは出荷時、あるいはその後において、図3に示す階層ツリー(木)構造における、自分のリーフからルートに至るまでのノードに割り当てられた鍵(ノードキー)および各リーフのリーフキーからなるキーセットをメモリに格納する。図3の最下段に示すK0000~K1111が各デバイス0~15にそれぞれ割り当てられたリーフキーであり、最上段のKR(ルートキー)から、最下段から2番目の節(ノード)に記載されたキー:KR~K111をノードキーとする。

【0051】図3に示すツリー構成において、例えばデバイス0はリーフキーK0000と、ノードキー:K000、K00、K0、KRを所有する。デバイス5はK0101、K010、K01、K0、KRを所有する。デバイス15は、K11111、K1111、K111、K11、KRを所有する。なお、図3のツリーにはデバイスが0~15の16個のみ記載され、ツリー構造も4段構成の均衡のとれた左右対称構成として示しているが、さらに多くのデバイスがツリー中に構成され、また、ツリーの各部において異なる段数構成を持つことが可能である。

【0052】また、図3のツリー構造に含まれる各デバイスには、様々な記録媒体、例えば、デバイス埋め込み型あるいはデバイスに着脱自在に構成されたDVD、CD、MD、フラッシュメモリ等を使用する様々なタイプのデバイスが含まれている。さらに、様々なアプリケーションサービスが共存可能である。このような異なるデバイス、異なるアプリケーションの共存構成の上に図3に示すコンテンツあるいは鍵配布構成である階層ツリー構造が適用される。

【0053】これらの様々なデバイス、アプリケーション

ンが共存するシステムにおいて、例えば図3の点線で囲んだ部分、すなわちデバイス0、1、2、3を同一の記録媒体を用いる1つのグループとして設定する。例えば、この点線で囲んだグループ内に含まれるデバイスに対しては、まとめて、共通のコンテンツを暗号化してプロバイダから送付したり、各デバイス共通に使用するコンテンツキーを送付したり、あるいは各デバイスからプロバイダあるいは決済機関等にコンテンツ料金の支払データをやはり暗号化して出力するといった処理が実行される。コンテンツプロバイダ、あるいは決済処理機関等、各デバイスとのデータ送受信を行なう機関は、図3の点線で囲んだ部分、すなわちデバイス0、1、2、3を1つのグループとして一括してデータを送付する処理を実行する。このようなグループは、図3のツリー中に複数存在する。コンテンツプロバイダ、あるいは決済処理機関等、各デバイスとのデータ送受信を行なう機関は、メッセージデータ配信手段として機能する。

【0054】なお、ノードキー、リーフキーは、ある1つの鍵管理センタによって統括して管理してもよいし、各グループに対する様々なデータ送受信を行なうプロバイダ、決済機関等のメッセージデータ配信手段によってグループごとに管理する構成としてもよい。これらのノードキー、リーフキーは例えばキーの漏洩等の場合に更新処理が実行され、この更新処理は鍵管理センタ、プロバイダ、決済機関等が実行する。

【0055】このツリー構造において、図3から明らかのように、1つのグループに含まれる3つのデバイス0、1、2、3はノードキーとして共通のキーK00、K0、KRを保有する。このノードキー共有構成を利用することにより、例えば共通のコンテンツキーをデバイス0、1、2、3のみに提供することが可能となる。たとえば、共通に保有するノードキーK00自体をコンテンツキーとして設定すれば、新たな鍵送付を実行することなくデバイス0、1、2、3のみが共通のコンテンツキーの設定が可能である。また、新たなコンテンツキーKconをノードキーK00で暗号化した値Enc(K00, Kcon)を、ネットワークを介してあるいは記録媒体に格納してデバイス0、1、2、3に配布すれば、デバイス0、1、2、3のみが、それぞれのデバイスにおいて保有する共有ノードキーK00を用いて暗号Enc(K00, Kcon)を解いてコンテンツキー:Kconを得ることが可能となる。なお、Enc(Ka, Kb)はKbをKaによって暗号化したデータであることを示す。

【0056】また、ある時点tにおいて、デバイス3の所有する鍵:K0011, K001, K00, K0, KRが攻撃者(ハッカー)により解析されて露呈したことが発覚した場合、それ以降、システム(デバイス0、1、2、3のグループ)で送受信されるデータを守るために、デバイス3をシステムから切り離す必要がある。そ

のためには、ノードキー：K 0 0 1, K 0 0, K 0, K R をそれぞれ新たな鍵 K (t) 0 0 1, K (t) 0 0, K (t) 0, K (t) R に更新し、デバイス 0, 1, 2 にその更新キーを伝える必要がある。ここで、K (t) a a a は、鍵 K a a a の世代 (Generation) : t の更新キーであることを示す。

【0057】更新キーの配布処理について説明する。キーの更新は、例えば、図 4 (A) に示す有効化キーブロック (E K B : Enabling Key Block) と呼ばれるブロックデータによって構成されるテーブルをたとえばネットワーク、あるいは記録媒体に格納してデバイス 0, 1, 2 に供給することによって実行される。なお、有効化キーブロック (E K B) は、図 3 に示すようなツリー構造を構成する各リーフに対応するデバイスに新たに更新されたキーを配布するための暗号化キーによって構成される。有効化キーブロック (E K B) は、キー更新ブロック (K R B : KeyRenewal Block) と呼ばれることもある。

【0058】図 4 (A) に示す有効化キーブロック (E K B) には、ノードキーの更新に必要なデバイスのみが更新可能なデータ構成を持つブロックデータとして構成される。図 4 の例は、図 3 に示すツリー構造中のデバイス 0, 1, 2 において、世代 t の更新ノードキーを配布することを目的として形成されたブロックデータである。図 3 から明らかなように、デバイス 0, デバイス 1 は、更新ノードキーとして K (t) 0 0, K (t) 0、K (t) R が必要であり、デバイス 2 は、更新ノードキーとして K (t) 0 0 1, K (t) 0 0, K (t) 0、K (t) R が必要である。

【0059】図 4 (A) の E K B に示されるように E K B には複数の暗号化キーが含まれる。最下段の暗号化キーは、E n c (K 0 0 1 0, K (t) 0 0 1) である。これはデバイス 2 の持つリーフキー K 0 0 1 0 によって暗号化された更新ノードキー K (t) 0 0 1 であり、デバイス 2 は、自身の持つリーフキーによってこの暗号化キーを復号し、K (t) 0 0 1 を得ることができる。また、復号により得た K (t) 0 0 1 を用いて、図 4

(A) の下から 2 段目の暗号化キー E n c (K (t) 0 0 1, K (t) 0 0) を復号可能となり、更新ノードキー K (t) 0 0 を得ることができる。以下順次、図 4 (A) の上から 2 段目の暗号化キー E n c (K (t) 0 0, K (t) 0) を復号し、更新ノードキー K (t) 0、図 4 (A) の上から 1 段目の暗号化キー E n c (K (t) 0, K (t) R) を復号し K (t) R を得る。一方、デバイス K 0 0 0 0, K 0 0 0 1 は、ノードキー K 0 0 0 は更新する対象に含まれておらず、更新ノードキーとして必要なのは、K (t) 0 0、K (t) 0、K (t) R である。デバイス K 0 0 0 0, K 0 0 0 1 は、図 4 (A) の上から 3 段目の暗号化キー E n c (K 0 0 0, K (t) 0 0) を復号し K (t) 0 0、を取得し、

以下、図 4 (A) の上から 2 段目の暗号化キー E n c (K (t) 0 0, K (t) 0) を復号し、更新ノードキー K (t) 0、図 4 (A) の上から 1 段目の暗号化キー E n c (K (t) 0, K (t) R) を復号し K (t) R を得る。このようにして、デバイス 0, 1, 2 は更新した鍵 K (t) 0 0 1, K (t) 0 0, K (t) 0, K (t) R を得ることができる。なお、図 4 (A) のインデックスは、復号キーとして使用するノードキー、リーフキーの絶対番地を示す。

【0060】図 3 に示すツリー構造の上位段のノードキー：K (t) 0, K (t) R の更新が不要であり、ノードキー K 0 0 のみの更新処理が必要である場合には、図 4 (B) の有効化キーブロック (E K B) を用いることで、更新ノードキー K (t) 0 0 をデバイス 0, 1, 2 に配布することができる。

【0061】図 4 (B) に示す E K B は、例えば特定のグループにおいて共有する新たなコンテンツキーを配布する場合に利用可能である。具体例として、図 3 に点線で示すグループ内のデバイス 0, 1, 2, 3 がある記録媒体を用いており、新たな共通のコンテンツキー K (t) c o n が必要であるとする。このとき、デバイス 0, 1, 2, 3 の共通のノードキー K 0 0 を更新した K (t) 0 0 を用いて新たな共通の更新コンテンツキー：K (t) c o n を暗号化したデータ E n c (K (t), K (t) c o n) を図 4 (B) に示す E K B とともに配布する。この配布により、デバイス 4 など、その他のグループの機器においては復号されないデータとしての配布が可能となる。

【0062】すなわち、デバイス 0, 1, 2 は E K B を処理して得た K (t) 0 0 を用いて上記暗号文を復号すれば、t 時点でのコンテンツキー K (t) c o n を得ることが可能になる。

【0063】[E K B を使用したコンテンツキーの配布] 図 5 に、t 時点でのコンテンツキー K (t) c o n を得る処理例として、K (t) 0 0 を用いて新たな共通のコンテンツキー K (t) c o n を暗号化したデータ E n c (K (t) 0 0, K (t) c o n) と図 4 (B) に示す E K B とを記録媒体を介して受領したデバイス 0 の処理を示す。すなわち E K B による暗号化メッセージデータをコンテンツキー K (t) c o n とした例である。

【0064】図 5 に示すように、デバイス 0 は、記録媒体に格納されている世代：t 時点の E K B と自分があらかじめ格納しているノードキー K 0 0 0 を用いて上述したと同様の E K B 処理により、ノードキー K (t) 0 0 を生成する。さらに、復号した更新ノードキー K (t) 0 0 を用いて更新コンテンツキー K (t) c o n を復号して、後にそれを使用するために自分だけが持つリーフキー K 0 0 0 0 で暗号化して格納する。

【0065】[E K B のフォーマット] 図 6 に有効化キーブロック (E K B) のフォーマット例を示す。ページ

ジョン601は、有効化キープロック(EKB)のバージョンを示す識別子である。なお、バージョンは最新のEKBを識別する機能とコンテンツとの対応関係を示す機能を持つ。デプスは、有効化キープロック(EKB)の配布先のデバイスに対する階層ツリーの階層数を示す。データポインタ603は、有効化キープロック(EKB)中のデータ部の位置を示すポインタであり、タグポインタ604はタグ部の位置、署名ポインタ605は署名の位置を示すポインタである。

【0066】データ部606は、例えば更新するノードキーを暗号化したデータを格納する。例えば図5に示すような更新されたノードキーに関する各暗号化キー等を格納する。

【0067】タグ部607は、データ部に格納された暗号化されたノードキー、リーフキーの位置関係を示すタグである。このタグの付与ルールを図7を用いて説明する。図7では、データとして先に図4(A)で説明した有効化キープロック(EKB)を送付する例を示している。この時のデータは、図7の表(b)に示すようになる。このときの暗号化キーに含まれるトップノードのアドレスをトップノードアドレスとする。この場合は、ルートキーの更新キーK(t)Rが含まれているので、トップノードアドレスはKRとなる。このとき、例えば最上段のデータEnc(K(t)0, K(t)R)は、図7の(a)に示す階層ツリーに示す位置にある。ここで、次のデータは、Enc(K(t)00, K(t)0)であり、ツリー上では前のデータの左下の位置にある。データがある場合は、タグが0、ない場合は1が設定される。タグは{左(L)タグ, 右(R)タグ}として設定される。最上段のデータEnc(K(t)0, K(t)R)の左にはデータがあるので、Lタグ=0、右にはデータがないので、Rタグ=1となる。以下、すべてのデータにタグが設定され、図7(c)に示すデータ列、およびタグ列が構成される。

【0068】タグは、データEnc(Kxxx, Kyyy)がツリー構造のどこに位置しているのかを示すために設定されるものである。データ部に格納されるキーデータEnc(Kxxx, Kyyy)...は、単純に暗号化されたキーの羅列データに過ぎないので、上述したタグによってデータとして格納された暗号化キーのツリー上の位置を判別可能としたものである。上述したタグを用いずに、先の図4で説明した構成のように暗号化データに対応させたノード・インデックスを用いて、例えば、

0: Enc(K(t)0, K(t)root)

00: Enc(K(t)00, K(t)0)

000: Enc(K(t)000, K(T)00)

...のようなデータ構成とすることも可能であるが、このようなインデックスを用いた構成とすると冗長なデータとなりデータ量が増大し、ネットワークを介する配

信等においては好ましくない。これに対し、上述したタグをキー位置を示す索引データとして用いることにより、少ないデータ量でキー位置の判別が可能となる。

【0069】図6に戻って、EKBフォーマットについてさらに説明する。署名(Signature)は、有効化キープロック(EKB)を発行した例えば鍵管理センタ、コンテンツロバイダ、決済機関等が実行する電子署名である。EKBを受領したデバイスは署名検証によって正当な有効化キープロック(EKB)発行者が発行した有効化キープロック(EKB)であることを確認する。

【0070】[EKBを使用したコンテンツキーおよびコンテンツの配信] 上述の例では、コンテンツキーのみをEKBとともに送付する例について説明したが、コンテンツキーで暗号化したコンテンツと、コンテンツキー暗号キーで暗号化したコンテンツキーと、EKBによって暗号化したコンテンツキー暗号鍵を併せて送付する構成について以下説明する。

【0071】図8にこのデータ構成を示す。図8(a)に示す構成において、Enc(Kcon, content)801は、コンテンツ(Content)をコンテンツキー(Kcon)で暗号化したデータであり、Enc(EK, Kcon)802は、コンテンツキー(Kcon)をコンテンツキー暗号キー(KEK: Key Encryption Key)で暗号化したデータであり、Enc(EKB, KEK)803は、コンテンツキー暗号キーKEKを有効化キープロック(EKB)によって暗号化したデータであることを示す。

【0072】ここで、コンテンツキー暗号キーKEKは、図3で示すノードキー(K000, K00...)、あるいはルートキー(KR)自体であってもよく、またノードキー(K000, K00...)、あるいはルートキー(KR)によって暗号化されたキーであってもよい。

【0073】図8(b)は、複数のコンテンツがメディアに記録され、それぞれが同じEnc(EKB, KEK)805を利用している場合の構成例を示す、このような構成においては、各データに同じEnc(EKB, KEK)を付加することなく、Enc(EKB, KEK)にリンクするリンク先を示すデータを各データに付加する構成とすることができる。

【0074】図9にコンテンツキー暗号キーKEKを、図3に示すノードキーK00を更新した更新ノードキーK(t)00として構成した場合の例を示す。この場合、図3の点線枠で囲んだグループにおいてデバイス3が、例えば鍵の漏洩によりリボーク(排除)されているとして、他のグループのメンバ、すなわち、デバイス0, 1, 2に対して図9に示す(a)有効化キープロック(EKB)と、(b)コンテンツキー(Kcon)をコンテンツキー暗号キー(KEK=K(t)00)で暗号化したデータと、(c)コンテンツ(content)をコンテンツキー(Kcon)で暗号化したデータとを配信

することにより、デバイス0, 1, 2はコンテンツを得ることができる。

【0075】図9の右側には、デバイス0における復号手順を示してある。デバイス0は、まず、受領した有効化キーブロックから自身の保有するリーフキー K_{000} を用いた復号処理により、コンテンツキー暗号キー($KEK = K(t)_{00}$)を取得する。次に、 $K(t)_{00}$ による復号によりコンテンツキー K_{con} を取得し、さらにコンテンツキー K_{con} によりコンテンツの復号を行なう。これらの処理により、デバイス0はコンテンツを利用可能となる。デバイス1, 2においても各々異なる処理手順で EKB を処理することにより、コンテンツキー暗号キー($KEK = K(t)_{00}$)を取得することが可能となり、同様にコンテンツを利用することが可能となる。

【0076】図3に示す他のグループのデバイス4, 5, 6…は、この同様のデータ(EKB)を受信したとしても、自身の保有するリーフキー、ノードキーを用いてコンテンツキー暗号キー($KEK = K(t)_{00}$)を取得することができない。同様にリポートされたデバイス3においても、自身の保有するリーフキー、ノードキーでは、コンテンツキー暗号キー($KEK = K(t)_{00}$)を取得することができず、正当な権利を有するデバイスのみがコンテンツを復号して利用することが可能となる。

【0077】このように、 EKB を利用したコンテンツキーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能とした暗号化コンテンツを配信することが可能となる。

【0078】なお、有効化キーブロック(EKB)、コンテンツキー、暗号化コンテンツ等は、ネットワークを介して安全に配信することが可能な構成であるが、有効化キーブロック(EKB)、コンテンツキー、暗号化コンテンツをDVD、CD等の記録媒体に格納してユーザに提供することも可能である。この場合、記録媒体に格納された暗号化コンテンツの復号には、同一の記録媒体に格納された有効化キーブロック(EKB)の復号により得られるコンテンツキーを使用するように構成すれば、予め正当権利者のみが保有するリーフキー、ノードキーによってのみ利用可能な暗号化コンテンツの配布処理、すなわち利用可能なユーザデバイスを限定したコンテンツ配布が簡易な構成で実現可能となる。

【0079】図10に記録媒体に暗号化コンテンツとともに有効化キーブロック(EKB)を格納した構成例を示す。図10に示す例においては、記録媒体にコンテンツC1~C4が格納され、さらに各格納コンテンツに対応する有効化キーブロック(EKB)を対応付けたデータが格納され、さらにバージョンMの有効化キーブロック(EKB_M)が格納されている。例えば EKB_1 はコンテンツC1を暗号化したコンテンツキー K_{co}

$n1$ を生成するのに使用され、例えば EKB_2 はコンテンツC2を暗号化したコンテンツキー K_{con2} を生成するのに使用される。この例では、バージョンMの有効化キーブロック(EKB_M)が記録媒体に格納されており、コンテンツC3, C4は有効化キーブロック(EKB_M)に対応付けられているので、有効化キーブロック(EKB_M)の復号によりコンテンツC3, C4のコンテンツキーを取得することができる。 EKB_1 , EKB_2 はディスクに格納されていないので、新たな提供手段、例えばネットワーク配信、あるいは記録媒体による配信によってそれぞれのコンテンツキーを復号するために必要な EKB_1 , EKB_2 を取得することが必要となる。

【0080】図11に、複数のデバイス間でコンテンツキーが流通する場合の EKB を利用したコンテンツキーの配信と、従来のコンテンツキー配信処理の比較例を示す。上段(a)従来構成であり、下段(b)が本発明の有効化キーブロック(EKB)を利用した例である。なお、図11において $K_a(K_b)$ は、 K_b を K_a で暗号化したデータであることを示す。

【0081】(a)に示すように、従来は、データ送信者の正当性を確認し、またデータ送信の暗号化処理に使用するセッションキー K_{ses} を共有するために各デバイス間において、認証処理および鍵交換処理(AK : Authentication and Key Exchange)を実行し、認証が成立したことを条件としてセッションキー K_{ses} でコンテンツキー K_{con} を暗号化して送信する処理を行っていた。

【0082】例えば図11(a)のPCにおいては、受信したセッションキーで暗号化したコンテンツキー $K_{ses}(K_{con})$ をセッションキーで復号して K_{con} を得ることが可能であり、さらに取得した K_{con} をPC自体の保有する保存キー K_{str} で暗号化して自身のメモリに保存することが可能となる。

【0083】図11(a)において、コンテンツプロバイダは、図11(a)の記録デバイス1101にのみデータを利用可能な形で配信したい場合でも、間にPC、再生装置が存在する場合は、図11(a)に示すように認証処理を実行し、それぞれのセッションキーでコンテンツキーを暗号化して配信するといった処理が必要となる。また、間に介在するPC、再生装置においても認証処理において生成し共有することになったセッションキーを用いることで暗号化コンテンツキーを復号してコンテンツキーを取得可能となる。

【0084】一方、図11(b)の下段に示す有効化キーブロック(EKB)を利用した例においては、コンテンツプロバイダから有効化キーブロック(EKB)と、有効化キーブロック(EKB)の処理によって得られるノードキー、またはルートキーによってコンテンツキー K_{con} を暗号化したデータ(図の例では K_{root}

(Kcon))を配信することにより、配信したEKBの処理が可能な機器においてのみコンテンツキーKconを復号して取得することが可能になる。

【0085】従って、例えば図11(b)の右端にのみ利用可能な有効化キープブロック(EKB)を生成して、その有効化キープブロック(EKB)と、そのEKB処理によって得られるノードキー、またはルートキーによってコンテンツキーKconを暗号化したデータを併せて送ることにより、間に存在するPC、再生機器等は、自身の有するリーフキー、ノードキーによっては、EKBの処理を実行することができない。従って、データ送受信デバイス間での認証処理、セッションキーの生成、セッションキーによるコンテンツキーKconの暗号化処理といった処理を実行することなく、安全に正当なデバイスに対してのみ利用可能なコンテンツキーを配信することが可能となる。

【0086】PC、記録再生器にも利用可能なコンテンツキーを配信したい場合は、それぞれにおいて処理可能な有効化キープブロック(EKB)を生成して、配信することにより、共通のコンテンツキーを取得することが可能となる。

【0087】[有効化キープブロック(EKB)を使用した認証キーの配信(共通鍵方式)] 上述の有効化キープブロック(EKB)を使用したデータあるいはキーの配信において、デバイス間で転送される有効化キープブロック(EKB)およびコンテンツあるいはコンテンツキーは常に同じ暗号化形態を維持しているため、データ伝走路を盗み出して記録し、再度、後で転送する、いわゆるリプレイアタックにより、不正コピーが生成される可能性がある。これを防ぐ構成としては、データ転送デバイス間において、従来と同様の認証処理および鍵交換処理を実行することが有効な手段である。ここでは、この認証処理および鍵交換処理を実行する際に使用する認証キーKakeを上述の有効化キープブロック(EKB)を使用してデバイスに配信することにより、安全な秘密鍵として共有する認証キーを持ち、共通鍵方式に従った認証処理を実行する構成について説明する。すなわちEKBによる暗号化メッセージデータを認証キーとした例である。

【0088】図12に、共通鍵暗号方式を用いた相互認証方法(ISO/IEC 9798-2)を示す。図12においては、共通鍵暗号方式としてDESを用いているが、共通鍵暗号方式であれば他の方式も可能である。図12において、まず、Bが64ビットの乱数Rbを生成し、Rbおよび自己のIDであるID(b)をAに送信する。これを受信したAは、新たに64ビットの乱数Raを生成し、Ra、Rb、ID(b)の順に、DESのCBCモードで鍵Kabを用いてデータを暗号化し、Bに返送する。なお、鍵Kabは、AおよびBに共通の秘密鍵としてそれぞれの記録素子内に格納する鍵である。DESの

CBCモードを用いた鍵Kabによる暗号化処理は、例えばDESを用いた処理においては、初期値とRaとを排他的論理和し、DES暗号化部において、鍵Kabを用いて暗号化し、暗号文E1を生成し、続けて暗号文E1とRbとを排他的論理和し、DES暗号化部において、鍵Kabを用いて暗号化し、暗号文E2を生成し、さらに、暗号文E2とID(b)とを排他的論理和し、DES暗号化部において、鍵Kabを用いて暗号化して生成した暗号文E3とによって送信データ(Token-AB)を生成する。

【0089】これを受信したBは、受信データを、やはり共通の秘密鍵としてそれぞれの記録素子内に格納する鍵Kab(認証キー)で復号化する。受信データの復号化方法は、まず、暗号文E1を認証キーKabで復号化し、乱数Raを得る。次に、暗号文E2を認証キーKabで復号化し、その結果とE1を排他的論理和し、Rbを得る。最後に、暗号文E3を認証キーKabで復号化し、その結果とE2を排他的論理和し、ID(b)を得る。こうして得られたRa、Rb、ID(b)のうち、RbおよびID(b)が、Bが送信したものと一致するか検証する。この検証に通った場合、BはAを正当なものとして認証する。

【0090】次にBは、認証後に使用するセッションキー(Kses)を生成する(生成方法は、乱数を用いる)。そして、Rb、Ra、Ksesの順に、DESのCBCモードで認証キーKabを用いて暗号化し、Aに返送する。

【0091】これを受信したAは、受信データを認証キーKabで復号化する。受信データの復号化方法は、Bの復号化処理と同様であるので、ここでは詳細を省略する。こうして得られたRb、Ra、Ksesの内、RbおよびRaが、Aが送信したものと一致するか検証する。この検証に通った場合、AはBを正当なものとして認証する。互いに相手を認証した後は、セッションキーKsesは、認証後の秘密通信のための共通鍵として利用される。

【0092】なお、受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして処理を中断する。

【0093】上述の認証処理においては、A、Bは共通の認証キーKabを共有する。この共通鍵Kabを上述の有効化キープブロック(EKB)を使用してデバイスに配信する。

【0094】例えば、図12の例では、A、またはBのいずれかが他方が復号可能な有効化キープブロック(EKB)を生成して生成した有効化キープブロック(EKB)によって認証キーKabを暗号化して、他方に送信する構成としてもよいし、あるいは第3者がデバイスA、Bに対して双方が利用可能な有効化キープブロック(EKB)を生成してデバイスA、Bに対して生成した有効化

キーブロック (EKB) によって認証キー K_{ab} を暗号化して配信する構成としてもよい。

【0095】図13および図14に複数のデバイスに共通の認証キー K_{ake} を有効化キーブロック (EKB) によって配信する構成例を示す。図13はデバイス0, 1, 2, 3に対して復号可能な認証キー K_{ake} を配信する例、図14はデバイス0, 1, 2, 3中のデバイス3をリポーク (排除) してデバイス0, 1, 2に対してのみ復号可能な認証キーを配信する例を示す。

【0096】図13の例では、更新ノードキー $K(t)$ 00によって、認証キー K_{ake} を暗号化したデータ (b) とともに、デバイス0, 1, 2, 3においてそれぞれの有するノードキー、リーフキーを用いて更新されたノードキー $K(t)$ 00を復号可能な有効化キーブロック (EKB) を生成して配信する。それぞれのデバイスは、図13の右側に示すようにまず、EKBを処理 (復号) することにより、更新されたノードキー $K(t)$ 00を取得し、次に、取得したノードキー $K(t)$ 00を用いて暗号化された認証キー: $Enc(K(t)00, K_{ake})$ を復号して認証キー K_{ake} を得ることが可能となる。

【0097】その他のデバイス4, 5, 6, 7…は同一の有効化キーブロック (EKB) を受信しても自身の保有するノードキー、リーフキーでは、EKBを処理して更新されたノードキー $K(t)$ 00を取得することができないので、安全に正当なデバイスに対してのみ認証キーを送付することができる。

【0098】一方、図14の例は、図3の点線枠で囲んだグループにおいてデバイス3が、例えば鍵の漏洩によりリポーク (排除) されているとして、他のグループのメンバ、すなわち、デバイス0, 1, 2, に対してのみ復号可能な有効化キーブロック (EKB) を生成して配信した例である。図14に示す (a) 有効化キーブロック (EKB) と、(b) 認証キー (K_{ake}) をノードキー ($K(t)$ 00) で暗号化したデータを配信する。

【0099】図14の右側には、復号手順を示してある。デバイス0, 1, 2は、まず、受領した有効化キーブロックから自身の保有するリーフキーまたはノードキーを用いた復号処理により、更新ノードキー ($K(t)$ 00) を取得する。次に、 $K(t)$ 00による復号により認証キー K_{ake} を取得する。

【0100】図3に示す他のグループのデバイス4, 5, 6…は、この同様のデータ (EKB) を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー ($K(t)$ 00) を取得することができない。同様にリポークされたデバイス3においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー ($K(t)$ 00) を取得することができず、正当な権利を有するデバイスのみが認証キーを復号して利用することが可能となる。

【0101】このように、EKBを利用した認証キーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能とした認証キーを配信することが可能となる。

【0102】〔公開鍵認証と有効化キーブロック (EKB) を使用したコンテンツキーの配信〕次に、公開鍵認証と有効化キーブロック (EKB) を使用したコンテンツキーの配信処理について説明する。まず、公開鍵暗号方式である160ビット長の楕円曲線暗号を用いた相互認証方法を、図15を用いて説明する。図15において、公開鍵暗号方式としてECCを用いているが、同様な公開鍵暗号方式であればいずれでもよい。また、鍵サイズも160ビットでなくてもよい。図15において、まずBが、64ビットの乱数 R_b を生成し、Aに送信する。これを受信したAは、新たに64ビットの乱数 R_a および素数 p より小さい乱数 k を生成する。そして、ベースポイント G を k 倍した点 $A_v = k \times G$ を求め、 R_a 、 R_b 、 A_v (X座標とY座標) に対する電子署名 A_{sig} を生成し、Aの公開鍵証明書とともにBに返送する。ここで、 R_a および R_b はそれぞれ64ビット、 A_v のX座標とY座標がそれぞれ160ビットであるので、合計448ビットに対する電子署名を生成する。

【0103】Aの公開鍵証明書、 R_a 、 R_b 、 A_v 、電子署名 A_{sig} を受信したBは、Aが送信してきた R_b が、Bが生成したものと一致するか検証する。その結果、一致していた場合には、Aの公開鍵証明書内の電子署名を認証局の公開鍵で検証し、Aの公開鍵を取り出す。そして、取り出したAの公開鍵を用い電子署名 A_{sig} を検証する。

【0104】次に、Bは、素数 p より小さい乱数 B_k を生成する。そして、ベースポイント G を B_k 倍した点 $B_v = B_k \times G$ を求め、 R_b 、 R_a 、 B_v (X座標とY座標) に対する電子署名 B_{sig} を生成し、Bの公開鍵証明書とともにAに返送する。

【0105】Bの公開鍵証明書、 R_b 、 R_a 、 A_v 、電子署名 B_{sig} を受信したAは、Bが送信してきた R_a が、Aが生成したものと一致するか検証する。その結果、一致していた場合には、Bの公開鍵証明書内の電子署名を認証局の公開鍵で検証し、Bの公開鍵を取り出す。そして、取り出したBの公開鍵を用い電子署名 B_{sig} を検証する。電子署名の検証に成功した後、AはBを正当なものとして認証する。

【0106】両者が認証に成功した場合には、Bは $B_k \times A_v$ (B_k は乱数だが、 A_v は楕円曲線上の点であるため、楕円曲線上の点のスカラー倍計算が必要) を計算し、Aは $A_k \times B_v$ を計算し、これら点のX座標の下位64ビットをセッションキーとして以降の通信に使用する (共通鍵暗号を64ビット鍵長の共通鍵暗号とした場合)。もちろん、Y座標からセッション鍵を生成しても

よいし、下位64ビットでなくてもよい。なお、相互認証後の秘密通信においては、送信データはセッションキーで暗号化されるだけでなく、電子署名も付されることがある。

【0107】電子署名の検証や受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして処理を中断する。

【0108】図16に公開鍵認証と有効化キープブロック(EKB)を使用したコンテンツキーの配信処理例を示す。まずコンテンツプロバイダとPC間において図15で説明した公開鍵方式による認証処理が実行される。コンテンツプロバイダは、コンテンツキー配信先である再生装置、記録媒体の有するノードキー、リーフキーによって復号可能なEKBを生成して、更新ノードキーによる暗号化を実行したコンテンツキーE(Kcon)と、有効化キープブロック(EKB)とをPC間の認証処理において生成したセッションキーKsesで暗号化してPCに送信する。

【0109】PCはセッションキーで暗号化された[更新ノードキーによる暗号化を実行したコンテンツキーE(Kcon)と、有効化キープブロック(EKB)]をセッションキーで復号した後、再生装置、記録媒体に送信する。

【0110】再生装置、記録媒体は、自身の保有するノードキーまたはリーフキーによって[更新ノードキーによる暗号化を実行したコンテンツキーE(Kcon)と、有効化キープブロック(EKB)]を復号することによってコンテンツキーKconを取得する。

【0111】この構成によれば、コンテンツプロバイダとPC間での認証を条件として[更新ノードキーによる暗号化を実行したコンテンツキーE(Kcon)と、有効化キープブロック(EKB)]が送信されるので、例えば、ノードキーの漏洩があった場合でも、確実な相手に対するデータ送信が可能となる。

【0112】[プログラムコードの有効化キープブロック(EKB)を使用した配信] 上述した例では、コンテンツキー、認証キー等を有効化キープブロック(EKB)を用いて暗号化して配信する方法を説明したが、様々なプログラムコードを有効化キープブロック(EKB)を用いて配信する構成も可能である。すなわちEKBによる暗号化メッセージデータをプログラムコードとした例である。以下、この構成について説明する。

【0113】図17にプログラムコードを有効化キープブロック(EKB)の例えば更新ノードキーによって暗号化してデバイス間で送信する例を示す。デバイス1701は、デバイス1702の有するノードキー、リーフキーによって復号可能な有効化キープブロック(EKB)と、有効化キープブロック(EKB)に含まれる更新ノードキーで暗号処理したプログラムコードをデバイス1702に送信する。デバイス1702は受信したEKBを

処理して更新ノードキーを取得して、さらに取得した更新ノードキーによってプログラムコードの復号を実行して、プログラムコードを得る。

【0114】図17に示す例では、さらに、デバイス1702において取得したプログラムコードによる処理を実行して、その結果をデバイス1701に返して、デバイス1701がその結果に基づいて、さらに処理を続行する例を示している。

【0115】このように有効化キープブロック(EKB)と、有効化キープブロック(EKB)に含まれる更新ノードキーで暗号処理したプログラムコードを配信することにより、特定のデバイスにおいて解読可能なプログラムコードを前述の図3で示した特定のデバイス、あるいはグループに対して配信することが可能となる。

【0116】[送信コンテンツに対するチェック値(ICV: Integrity Check Value)を対応させる構成] 次に、コンテンツの改竄を防止するためにコンテンツのインテグリティ・チェック値(ICV)を生成して、コンテンツに対応付けて、ICVの計算により、コンテンツ改竄の有無を判定する処理構成について説明する。

【0117】コンテンツのインテグリティ・チェック値(ICV)は、例えばコンテンツに対するハッシュ関数を用いて計算され、 $ICV = hash(Kicv, C1, C2, \dots)$ によって計算される。KicvはICV生成キーである。C1, C2はコンテンツの情報であり、コンテンツの重要情報のメッセージ認証符号(MAC: Message authentication Code)が使用される。

【0118】DES暗号処理構成を用いたMAC値生成例を図18に示す。図18の構成に示すように対象となるメッセージを8バイト単位に分割し、(以下、分割されたメッセージをM1、M2、・・・、MNとする)、まず、初期値(Initial Value(以下、IVとする))とM1を排他的論理和する(その結果をI1とする)。次に、I1をDES暗号化部に入れ、鍵(以下、K1とする)を用いて暗号化する(出力をE1とする)。続けて、E1およびM2を排他的論理和し、その出力I2をDES暗号化部へ入れ、鍵K1を用いて暗号化する(出力E2)。以下、これを繰り返し、全てのメッセージに対して暗号化処理を施す。最後に出てきたENがメッセージ認証符号(MAC(Message Authentication Code))となる。

【0119】このようなコンテンツのMAC値とICV生成キーにハッシュ関数を適用して用いてコンテンツのインテグリティ・チェック値(ICV)が生成される。改竄のないことが保証された例えばコンテンツ生成時に生成したICVと、新たにコンテンツに基づいて生成したICVとを比較して同一のICVが得られればコンテンツに改竄のないことが保証され、ICVが異なれば、改竄があったと判定される。

【0120】[チェック値(ICV)の生成キーKic

vをEKBによって配布する構成] 次に、コンテンツのインテグリティ・チェック値(ICV)生成キーであるKicvを上記の有効化キープブロックによって送付する構成について説明する。すなわちEKBによる暗号化メッセージデータをコンテンツのインテグリティ・チェック値(ICV)生成キーとした例である。

【0121】図19および図20に複数のデバイスに共通のコンテンツを送付した場合、それらのコンテンツの改竄の有無を検証するためのインテグリティ・チェック値生成キーKicvを有効化キープブロック(EKB)によって配信する構成例を示す。図19はデバイス0, 1, 2, 3に対して復号可能なチェック値生成キーKicvを配信する例、図20はデバイス0, 1, 2, 3中のデバイス3をリボーク(排除)してデバイス0, 1, 2に対してのみ復号可能なチェック値生成キーKicvを配信する例を示す。

【0122】図19の例では、更新ノードキーK(t)00によって、チェック値生成キーKicvを暗号化したデータ(b)とともに、デバイス0, 1, 2, 3においてそれぞれの有するノードキー、リーフキーを用いて更新されたノードキーK(t)00を復号可能な有効化キープブロック(EKB)を生成して配信する。それぞれのデバイスは、図19の右側に示すようにまず、EKBを処理(復号)することにより、更新されたノードキーK(t)00を取得し、次に、取得したノードキーK(t)00を用いて暗号化されたチェック値生成キー: Enc(K(t)00, Kicv)を復号してチェック値生成キーKicvを得ることが可能となる。

【0123】その他のデバイス4, 5, 6, 7…は同一の有効化キープブロック(EKB)を受信しても自身の保有するノードキー、リーフキーでは、EKBを処理して更新されたノードキーK(t)00を取得することができないので、安全に正当なデバイスに対してのみチェック値生成キーを送付することができる。

【0124】一方、図20の例は、図3の点線枠で囲んだグループにおいてデバイス3が、例えば鍵の漏洩によりリボーク(排除)されているとして、他のグループのメンバ、すなわち、デバイス0, 1, 2, に対してのみ復号可能な有効化キープブロック(EKB)を生成して配信した例である。図20に示す(a)有効化キープブロック(EKB)と、(b)チェック値生成キー(Kicv)をノードキー(K(t)00)で暗号化したデータを配信する。

【0125】図20の右側には、復号手順を示してある。デバイス0, 1, 2は、まず、受領した有効化キープブロックから自身の保有するリーフキーまたはノードキーを用いた復号処理により、更新ノードキー(K(t)00)を取得する。次に、K(t)00による復号によりチェック値生成キーKicvを取得する。

【0126】図3に示す他のグループのデバイス4,

5, 6…は、この同様のデータ(EKB)を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー(K(t)00)を取得することができない。同様にリボークされたデバイス3においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー(K(t)00)を取得することができず、正当な権利を有するデバイスのみがチェック値生成キーを復号して利用することが可能となる。

【0127】このように、EKBを利用したチェック値生成キーの配送を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能としたチェック値生成キーを配信することが可能となる。

【0128】このようなコンテンツのインテグリティ・チェック値(ICV)を用いることにより、EKBと暗号化コンテンツの不正コピーを排除することができる。例えば図21に示すように、コンテンツC1とコンテンツC2とをそれぞれのコンテンツキーを取得可能な有効化キープブロック(EKB)とともに格納したメディア1があり、これをそのままメディア2にコピーした場合を想定する。EKBと暗号化コンテンツのコピーは可能であり、これをEKBを復号可能なデバイスでは利用することになる。

【0129】図21の(b)に示すように各メディアに正当に格納されたコンテンツに対応付けてインテグリティ・チェック値(ICV(C1, C2))を格納する構成とする。なお、(ICV(C1, C2))は、コンテンツC1とコンテンツC2にハッシュ関数を用いて計算されるコンテンツのインテグリティ・チェック値である $ICV = hash(Kicv, C1, C2)$ を示している。図21の(b)の構成において、メディア1には正当にコンテンツ1とコンテンツ2が格納され、コンテンツC1とコンテンツC2に基づいて生成されたインテグリティ・チェック値(ICV(C1, C2))が格納される。また、メディア2には正当にコンテンツ1が格納され、コンテンツC1に基づいて生成されたインテグリティ・チェック値(ICV(C1))が格納される。この構成において、メディア1に格納された{EKB, コンテンツ2}をメディア2にコピーしたとすると、メディア2で、コンテンツチェック値を新たに生成するとICV(C1, C2)が生成されることになり、メディアに格納されているKicv(C1)と異なり、コンテンツの改竄あるいは不正なコピーによる新たなコンテンツの格納が実行されたことが明らかになる。メディアを再生するデバイスにおいて、再生ステップの前ステップにICVチェックを実行して、生成ICVと格納ICVの一致を判別し、一致しない場合は、再生を実行しない構成とすることにより、不正コピーのコンテンツの再生を防止することが可能となる。

【0130】また、さらに、安全性を高めるため、コンテンツのインテグリティ・チェック値(ICV)を書き

換えカウンタを含めたデータに基づいて生成する構成としてもよい。すなわち $ICV = hash(Kicv, counter+1, C1, C2, \dots)$ によって計算する構成とする。ここで、カウンタ ($counter+1$) は、 ICV の書き換えごとに1つインクリメントされる値として設定する。なお、カウンタ値はセキュアメモリに格納する構成とすることが必要である。

【0131】さらに、コンテンツのインテグリティ・チェック値 (ICV) をコンテンツと同一メディアに格納することができない構成においては、コンテンツのインテグリティ・チェック値 (ICV) をコンテンツとは別のメディア上に格納する構成としてもよい。

【0132】例えば、読み込み専用メディアや通常のMO等のコピー防止策のとられていないメディアにコンテンツを格納する場合、同一メディアにインテグリティ・チェック値 (ICV) を格納すると ICV の書き換えが不正なユーザによりなされる可能性があり、 ICV の安全性が保てないおそれがある。この様な場合、ホストマシン上の安全なメディアに ICV を格納して、コンテンツのコピーコントロール (例えば $check-in/check-out$ 、 $move$) に ICV を使用する構成とすることにより、 ICV の安全な管理およびコンテンツの改竄チェックが可能となる。

【0133】この構成例を図22に示す。図22では読み込み専用メディアや通常のMO等のコピー防止策のとられていないメディア2201にコンテンツが格納され、これらのコンテンツに関するインテグリティ・チェック値 (ICV) を、ユーザが自由にアクセスすることの許可されないホストマシン上の安全なメディア2202に格納し、ユーザによる不正なインテグリティ・チェック値 (ICV) の書き換えを防止した例である。このような構成として、例えばメディア2201を装着したデバイスがメディア2201の再生を実行する際にホストマシンであるPC、サーバにおいて ICV のチェックを実行して再生の可否を判定する構成とすれば、不正なコピーコンテンツあるいは改竄コンテンツの再生を防止できる。

【0134】[階層ツリー構造のカテゴリー分類] 暗号鍵をルートキー、ノードキー、リーフキー等、図3の階層ツリー構造として構成し、コンテンツキー、認証キー、 ICV 生成キー、あるいはプログラムコード、データ等を有効化キーブロック (EKB) とともに暗号化して配信する構成について説明してきたが、ノードキー等を定義している階層ツリー構造を各デバイスのカテゴリー毎に分類して効率的なキー更新処理、暗号化キー配信、データ配信を実行する構成について、以下説明する。

【0135】図23に階層ツリー構造のカテゴリーの分類の一例を示す。図23において、階層ツリー構造の最上段には、ルートキー $Kroot2301$ が設定され、

以下の中間段にはノードキー2302が設定され、最下段には、リーフキー2303が設定される。各デバイスは個々のリーフキーと、リーフキーからルートキーに至る一連のノードキー、ルートキーを保有する。

【0136】ここで、一例として最上段から第M段目のあるノードをカテゴリノード2304として設定する。すなわち第M段目のノードの各々を特定カテゴリのデバイス設定ノードとする。第M段の1つのノードを頂点として以下、 $M+1$ 段以下のノード、リーフは、そのカテゴリに含まれるデバイスに関するノードおよびリーフとする。

【0137】例えば図23の第M段目の1つのノード2305にはカテゴリ [メモリスティック (商標)] が設定され、このノード以下に連なるノード、リーフはメモリスティックを使用した様々なデバイスを含むカテゴリ専用のノードまたはリーフとして設定される。すなわち、ノード2305以下を、メモリスティックのカテゴリに定義されるデバイスの関連ノード、およびリーフの集合として定義する。

【0138】さらに、M段から数段分下位の段をサブカテゴリノード2306として設定することができる。例えば図に示すようにカテゴリ [メモリスティック] ノード2305の2段下のノードに、メモリスティックを使用したデバイスのカテゴリに含まれるサブカテゴリノードとして、[再生専用器] のノードを設定する。さらに、サブカテゴリノードである再生専用器のノード2306以下に、再生専用器のカテゴリに含まれる音楽再生機能付き電話のノード2307が設定され、さらにその下位に、音楽再生機能付き電話のカテゴリに含まれる

[PHS] ノード2308と [携帯電話] ノード2309を設定することができる。

【0139】さらに、カテゴリ、サブカテゴリは、デバイスの種類のみならず、例えばあるメーカー、コンテンツプロバイダ、決済機関等が独自に管理するノード、すなわち処理単位、管轄単位、あるいは提供サービス単位等、任意の単位 (これらを総称して以下、エンティティと呼ぶ) で設定することが可能である。例えば1つのカテゴリノードをゲーム機器メーカーの販売するゲーム機器XYZ専用の頂点ノードとして設定すれば、メーカーの販売するゲーム機器XYZにその頂点ノード以下の下段のノードキー、リーフキーを格納して販売することが可能となり、その後、暗号化コンテンツの配信、あるいは各種キーの配信、更新処理を、その頂点ノードキー以下のノードキー、リーフキーによって構成される有効化キーブロック (EKB) を生成して配信し、頂点ノード以下のデバイスに対してのみ利用可能なデータが配信可能となる。

【0140】このように、1つのノードを頂点として、以下のノードをその頂点ノードに定義されたカテゴリ、あるいはサブカテゴリの関連ノードとして設定する

構成とすることにより、カテゴリ段、あるいはサブカテゴリ段の1つの頂点ノードを管理するメーカー、コンテンツプロバイダ等がそのノードを頂点とする有効化キーブロック (EKB) を独自に生成して、頂点ノード以下に属するデバイスに配信する構成が可能となり、頂点ノードに属さない他のカテゴリのノードに属するデバイスには全く影響を及ぼさずにキー更新を実行することができる。

【0141】〔簡略EKBによるキー配信構成〕先に説明した例えば図3のツリー構成において、キー、例えばコンテンツキーを所定デバイス (リーフ) 宛に送付する場合、キー配布先デバイスの所有しているリーフキー、ノードキーを用いて復号可能な有効化キーブロック (EKB) を生成して提供する。例えば図24 (a) に示すツリー構成において、リーフを構成するデバイス a, g, j に対してキー、例えばコンテンツキーを送信する場合、a, g, j の各ノードにおいて復号可能な有効化キーブロック (EKB) を生成して配信する。

【0142】例えば更新ルートキー $K(t)_{root}$ でコンテンツキー $K(t)_{con}$ を暗号化処理し、EKBとともに配信する場合を考える。この場合、デバイス a, g, j は、それぞれが図24 (b) に示すリーフおよびノードキーを用いて、EKBの処理を実行して $K(t)_{root}$ を取得し、取得した更新ルートキー $K(t)_{root}$ によってコンテンツキー $K(t)_{con}$ の復号処理を実行してコンテンツキーを得る。

【0143】この場合に提供される有効化キーブロック (EKB) の構成は、図25に示すようになる。図25に示す有効化キーブロック (EKB) は、先の図6で説明した有効化キーブロック (EKB) のフォーマットにしたがって構成されたものであり、データ (暗号化キー) と対応するタグを持つ。タグは、先に図7を用いて説明したように左 (L)、右 (R)、それぞれの方向にデータがあれば0、無ければ1を示している。

【0144】有効化キーブロック (EKB) を受領したデバイスは、有効化キーブロック (EKB) の暗号化キーとタグに基づいて、順次暗号化キーの復号処理を実行して上位ノードの更新キーを取得していく。図25に示すように、有効化キーブロック (EKB) は、ルートからリーフまでの段数 (デプス) が多いほど、そのデータ量は増加していく。段数 (デプス) は、デバイス (リーフ) 数に応じて増大するものであり、キーの配信先となるデバイス数が多い場合は、EKBのデータ量がさらに増大することになる。

【0145】このような有効化キーブロック (EKB) のデータ量の削減を可能とした構成について説明する。図26は、有効化キーブロック (EKB) をキー配信デバイスに応じて簡略化して構成した例を示すものである。

【0146】図25と同様、リーフを構成するデバイス

a, g, j に対してキー、例えばコンテンツキーを送信する場合を想定する。図26の (a) に示すように、キー配信デバイスによってのみ構成されるツリーを構築する。この場合、図24 (b) に示す構成に基づいて新たなツリー構成として図26 (b) のツリー構成が構築される。KrootからKjまでは全く分岐がなく1つの枝のみが存在すればよく、KrootからKaおよびKgに至るためには、K0に分岐点を構成するのみで、2分岐構成の図26 (a) のツリーが構築される。

【0147】図26 (a) に示すように、ノードとしてK0のみを持つ簡略化したツリーが生成される。更新キー配信のための有効化キーブロック (EKB) は、これらの簡略ツリーに基づいて生成する。図26 (a) に示すツリーは、有効化キーブロック (EKB) を復号可能な末端ノードまたはリーフを最下段とした2分岐型ツリーを構成するパスを選択して不要ノードを省略することにより再構築される再構築階層ツリーである。更新キー配信のための有効化キーブロック (EKB) は、この再構築階層ツリーのノードまたはリーフに対応するキーのみに基づいて構成される。

【0148】先の図25で説明した有効化キーブロック (EKB) は、各リーフ a, g, j からKrootに至るまでのすべてのキーを暗号化したデータを格納していたが、簡略化EKBは、簡略化したツリーを構成するノードについてのみの暗号化データを格納する。図26

(b) に示すようにタグは3ビット構成を有する。第1および第2ビットは、図25の例と、同様の意味を持ち、左 (L)、右 (R)、それぞれの方向にデータがあれば0、無ければ1を示す。第3番目のビットは、EKB内に暗号化キーが格納されているか否かを示すためのビットであり、データが格納されている場合は1、データが無い場合は、0として設定される。

【0149】データ通信網、あるいは記憶媒体に格納されてデバイス (リーフ) に提供される有効化キーブロック (EKB) は、図26 (b) に示すように、図25に示す構成と比較すると、データ量が大幅に削減されたものとなる。図26に示す有効化キーブロック (EKB) を受領した各デバイスは、タグの第3ビットに1が格納された部分のデータのみを順次復号することにより、所定の暗号化キーの復号を実現することができる。例えばデバイス a は、暗号化データ $Enc(Ka, K(t)0)$ をリーフキー Ka で復号して、ノードキー $K(t)0$ を取得して、ノードキー $K(t)0$ によって暗号化データ $Enc(K(t)0, K(t)_{root})$ を復号して $K(t)_{root}$ を取得する。デバイス j は、暗号化データ $Enc(Kj, K(t)_{root})$ をリーフキー Kj で復号して、 $K(t)_{root}$ を取得する。

【0150】このように、配信先のデバイスによってのみ構成される簡略化した新たなツリー構成を構築して、構築されたツリーを構成するリーフおよびノードのキー

のみを用いて有効化キーブロック (EKB) を生成することにより、少ないデータ量の有効化キーブロック (EKB) を生成することが可能となり、有効化キーブロック (EKB) のデータ配信が効率的に実行可能となる。

【0151】なお、簡略化した階層ツリー構成は、後段で説明するエンティティ単位のEKB管理構成において特に有効に活用可能である。エンティティは、キー配信構成としてのツリー構成を構成するノードあるいはリーフから選択した複数のノードあるいはリーフの集合体ブロックである。エンティティは、デバイスの種類に応じて設定される集合であったり、あるいはデバイス提供メーカー、コンテンツプロバイダ、決済機関等の管理単位等、ある共通点を持った処理単位、管轄単位、あるいは提供サービス単位等、様々な態様の集合として設定される。1つのエンティティには、ある共通のカテゴリに分類されるデバイスが集まっており、例えば複数のエンティティの頂点ノード (サブルート) によって上述したと同様の簡略化したツリーを再構築してEKBを生成することにより、選択されたエンティティに属するデバイスにおいて復号可能な簡略化された有効化キーブロック (EKB) の生成、配信が可能となる。エンティティ単位の管理構成については後段で詳細に説明する。

【0152】なお、このような有効化キーブロック (EKB) は、光ディスク、DVD等の情報記録媒体に格納した構成とすることが可能である。例えば、上述の暗号化キーデータによって構成されるデータ部と、暗号化キーデータの階層ツリー構造における位置識別データとしてのタグ部とを含む有効化キーブロック (EKB) にさらに、更新ノードキーによって暗号化したコンテンツ等のメッセージデータとを格納した情報記録媒体を各デバイスに提供する構成が可能である。デバイスは有効化キーブロック (EKB) に含まれる暗号化キーデータをタグ部の識別データにしたがって順次抽出して復号し、コンテンツの復号に必要なキーを取得してコンテンツの利用を行なうことが可能となる。もちろん、有効化キーブロック (EKB) をインターネット等のネットワークを介して配信する構成としてもよい。

【0153】[エンティティ単位のEKB管理構成] 次に、キー配信構成としてのツリー構成を構成するノードあるいはリーフを、複数のノードあるいはリーフの集合としてのブロックで管理する構成について説明する。なお、複数のノードあるいはリーフの集合としてのブロックを以下エンティティと呼ぶ。エンティティは、デバイスの種類に応じて設定される集合であったり、あるいはデバイス提供メーカー、コンテンツプロバイダ、決済機関等の管理単位等、ある共通点を持った処理単位、管轄単位、あるいは提供サービス単位等、様々な態様の集合として設定される。すなわち、エンティティは、デバイス種類、サービス種類、管理手段種類等の共通のカテゴリに属するデバイスあるいはエンティティの管理主体と

して定義される。

【0154】エンティティについて、図27を用いて説明する。図27(a)はツリーのエンティティ単位での管理構成を説明する図である。1つのエンティティは図では、三角形として示し、例えば1エンティティ2701内には、複数のノードが含まれる。1エンティティ内のノード構成を示すのが(b)である。1つのエンティティは、1つのノードを頂点とした複数段の2分岐形ツリーによって構成される。以下、エンティティの頂点ノード2702をサブルートと呼ぶ。

【0155】ツリーの末端は、(c)に示すようにリーフ、すなわちデバイスによって構成される。デバイスは、複数デバイスをリーフとし、サブルートである頂点ノード2702を持つツリーによって構成されるいずれかのエンティティに属する。

【0156】図27(a)から理解されるように、エンティティは、階層構造を持つ。この階層構造について、図28を用いて説明する。

【0157】図28(a)は、階層構造を簡略化して説明するための図であり、K r o o tから数段下の段にエンティティA01~Annが構成され、エンティティA1~Anの下位には、さらに、エンティティB01~Bnk、さらに、その下位にエンティティC1~Cnqが設定されている。各エンティティは、図28(b)、

(c)に示す如く、複数段のノード、リーフによって構成されるツリー形状を持つ。

【0158】例えばエンティティBnkの構成は、

(b)に示すように、サブルート2811を頂点ノードとして、末端ノード2812に至るまでの複数ノードを有する。このエンティティは識別子Bnkを持ち、エンティティBnk内のノードに対応するノードキー管理をエンティティBnk独自に実行することにより、末端ノード2812を頂点として設定される下位(子)エンティティの管理を実行する。また、一方、エンティティBnkは、サブルート2811を末端ノードとして持つ上位(親)エンティティAnnの管理下にある。

【0159】エンティティCn3の構成は、(c)に示すように、サブルート2851を頂点ノードとして、各デバイスである末端ノード2852、この場合はリーフに至るまで複数ノード、リーフを有する。このエンティティは識別子Cn3を持ち、エンティティCn3内のノード、リーフに対応するノードキー、リーフキー管理をエンティティCn3独自に実行することにより、末端ノード2852に対応するリーフ(デバイス)の管理を実行する。また、一方、エンティティCn3は、サブルート2851を末端ノードとして持つ上位(親)エンティティBn2の管理下にある。各エンティティにおけるキー管理とは、例えばキー更新処理、リボーク処理等であるが、これらは後段で詳細に説明する。

【0160】最下段エンティティのリーフであるデバイ

スには、デバイスの属するエンティティのリーフキーから、自己の属するエンティティの頂点ノードであるサブルートノードに至るパスに位置する各ノードのノードキーおよびリーフキーが格納される。例えば末端ノード2852のデバイスは、末端ノード（リーフ）2852から、サブルートノード2851までの各キーを格納する。

【0161】図29を用いて、さらにエンティティの構成について説明する。エンティティは様々な段数によって構成されるツリー構造を持つことが可能である。段数、すなわちデプス（depth）は、エンティティで管理する末端ノードに対応する下位（子）エンティティの数、あるいはリーフとしてのデバイス数に応じて設定可能である。

【0162】図29の（a）に示すような上下エンティティ構成を具体化すると、（b）に示す態様となる。ルートエンティティは、ルートキーを持つ最上段のエンティティである。ルートエンティティの末端ノードに複数の下位エンティティとしてエンティティA、B、Cが設定され、さらに、エンティティCの下位エンティティとしてエンティティDが設定される。エンティティCは2901は、その末端ノードの1つ以上のノードをリザーブノード2950として保持し、自己の管理するエンティティを増加させる場合、さらに複数段のツリー構成を持つエンティティC'2902をリザーブノード2950を頂点ノードとして新設することにより、管理末端ノード2970を増加させて、管理末端ノードに増加した下位エンティティを追加することができる。

【0163】リザーブノードについて、さらに図30を用いて説明する。エンティティA、3011は、管理する下位エンティティB、C、D…を持ち、1つのリザーブノード3021を持つ。エンティティは管理対象の下位エンティティをさらに増加させたい場合、リザーブノード3021に、自己管理の下位エンティティA'、3012を設定し、下位エンティティA'、3012の末端ノードにさらに管理対象の下位エンティティF、Gを設定することができる。自己管理の下位エンティティA'、3012も、その末端ノードの少なくとも1つをリザーブノード3022として設定することにより、さらに下位エンティティA''3013を設定して、さらに管理エンティティを増加させることができる。下位エンティティA''3013の末端ノードにも1以上のリザーブノードを確保する。このようなリザーブノード保有構成をとることにより、あるエンティティの管理する下位エンティティは、際限なく増加させることが可能となる。なお、リザーブエンティティは、末端ノードの1つのみではなく、複数個設定する構成としてもよい。

【0164】それぞれのエンティティでは、エンティティ単位で有効化キープブロック（EKB）が構成され、エンティティ単位でのキー更新、リボーク処理を実行する

ことになる。図30のように複数のエンティティA、A'、A''には各エンティティ個々の有効化キープブロック（EKB）が設定されることになるが、これらは、エンティティA、A'、A''を共通に管理する例えばあるデバイスメーカーが一括して管理することが可能である。

【0165】[新規エンティティの登録処理] 次に、新規エンティティの登録処理について説明する。登録処理シーケンスを図31に示す。図31のシーケンスにしたがって説明する。新たにツリー構成中に追加される新規（子）エンティティ（N-E n）は、上位（親）エンティティ（P-E n）に対して新規登録要求を実行する。なお、各エンティティは、公開鍵暗号方式に従った公開鍵を保有し、新規エンティティは自己の公開鍵を登録要求に際して上位エンティティ（P-E n）に送付する。

【0166】登録要求を受領した上位エンティティ（P-E n）は、受領した新規（子）エンティティ（N-E n）の公開鍵を証明書発行局（CA：Certificate Authority）に転送し、CAの署名を付加した新規（子）エンティティ（N-E n）の公開鍵を受領する。これらの手続きは、上位エンティティ（P-E n）と新規（子）エンティティ（N-E n）との相互認証の手続きとして行われる。

【0167】これらの処理により、新規登録要求エンティティの認証が終了すると、上位エンティティ（P-E n）は、新規（子）エンティティ（N-E n）の登録を許可し、新規（子）エンティティ（N-E n）のノードキーを新規（子）エンティティ（N-E n）に送信する。このノードキーは、上位エンティティ（P-E n）の末端ノードの1つのノードキーであり、かつ、新規（子）エンティティ（N-E n）の頂点ノード、すなわちサブルートキーに対応する。

【0168】このノードキー送信が終了すると、新規（子）エンティティ（N-E n）は、新規（子）エンティティ（N-E n）のツリー構成を構築し、構築したツリーの頂点に受信した頂点ノードのサブルートキーを設定し、各ノード、リーフのキーを設定して、エンティティ内の有効化キープブロック（EKB）を生成する。1つのエンティティ内の有効化キープブロック（EKB）をサブEKBと呼ぶ。

【0169】一方、上位エンティティ（P-E n）は、新規（子）エンティティ（N-E n）の追加により、有効化する末端ノードを追加した上位エンティティ（P-E n）内のサブEKBを生成する。

【0170】新規（子）エンティティ（N-E n）は、新規（子）エンティティ（N-E n）内のノードキー、リーフキーによって構成されるサブEKBを生成すると、これを上位エンティティ（P-E n）に送信する。

【0171】新規（子）エンティティ（N-E n）からサブEKBを受信した上位エンティティ（P-E n）

は、受信したサブEKBと、上位エンティティ（P-E n）の更新したサブEKBとをキー発行センター（KDC：Key Distribute Center）に送信する。

【0172】キー発行センター（KDC）は、すべてのエンティティのサブEKBに基づいて、様々な態様のEKB、すなわち特定のエンティティあるいはデバイスのみが復号可能なEKBを生成することが可能となる。このように復号可能なエンティティあるいはデバイスを設定したEKBを例えばコンテンツプロバイダに提供し、コンテンツプロバイダがEKBに基づいてコンテンツキーを暗号化して、ネットワークを介して、あるいは記録媒体に格納して提供することにより、特定のデバイスでのみ利用可能なコンテンツを提供することが可能となる。

【0173】なお、新規エンティティのサブEKBのキー発行センター（KDC）に対する登録処理は、サブEKBを上位エンティティを介してを順次転送して実行する方法に限るものではなく、上位エンティティを介さずに、新規登録エンティティから直接、キー発行センター（KDC）に登録する処理を実行する構成としてもよい。

【0174】上位エンティティと、上位エンティティに新規追加する下位エンティティとの対応について図32を用いて説明する。上位エンティティの末端ノードの1つ3201を新規追加エンティティの頂点ノードとして、下位エンティティに提供することによって下位エンティティは、上位エンティティの管理下のエンティティとして追加される。上位エンティティの管理下のエンティティとは、後段で詳細に説明するが、下位エンティティのリボーク（排除）処理を上位エンティティが実行できる構成であるという意味を含むものである。

【0175】図32に示すように、上位エンティティに新規エンティティが設定されると、上位エンティティのリーフである末端ノードの1つのノード3201と新規追加エンティティの頂点ノード3202とが等しいノードとして設定される。すなわち上位ノードの1つのリーフである1つの末端ノードが、新規追加エンティティのサブルートとして設定される。このように設定されることにより、新規追加エンティティが全体ツリー構成の下で有効化される。

【0176】図33に新規追加エンティティを設定した際に上位エンティティが生成する更新EKBの例を示す。図33は、（a）に示す構成、すなわち既に有効に存在する末端ノード（node000）3301と末端ノード（node001）3302があり、ここに新規追加エンティティに新規エンティティ追加末端ノード（node100）3303を付与した際に上位エンティティが生成するサブEKBの例を示したものである。

【0177】サブEKBは、図33の（b）に示すような構成を持つ。それぞれ有効に存在する末端ノードキ

ーにより暗号化された上位ノードキー、上位ノードキーで暗号化されたさらなる上位ノードキー、…さらに上位に進行してサブルートキーに至る構成となっている。この構成によりサブEKBが生成される。各エンティティは図33（b）に示すと同様、有効な末端ノード、あるいはリーフキーにより暗号化された上位ノードキー、上位ノードキーでさらに上位のノードキーを暗号化し、順次上位に深層してサブルートに至る暗号化データによって構成されるEKBを有し、これを管理する。

【0178】〔エンティティ管理下におけるリボーク処理〕次に、キー配信ツリー構成をエンティティ単位として管理する構成におけるデバイスあるいはエンティティのリボーク（排除）処理について説明する。先の図3、4では、ツリー構成全体の中から特定のデバイスのみ復号可能で、リボークされたデバイスは復号不可能な有効化キープロック（EKB）を配信する処理について説明した。図3、4で説明したリボーク処理は、ツリー全体の中から特定のリーフであるデバイスをリボークする処理であったが、ツリーのエディティ管理による構成では、エンティティ毎にリボーク処理が実行可能となる。

【0179】図34以下の図を用いてエンティティ管理下のツリー構成におけるリボーク処理について説明する。図34は、ツリーを構成するエンティティのうち、最下段のエンティティ、すなわち個々のデバイスを管理しているエンティティによるデバイスのリボーク処理を説明する図である。

【0180】図34（a）は、エンティティ管理によるキー配信ツリー構造を示している。ツリー最上位にはルートノードが設定され、その数段下にエンティティA01～Ann、さらにその下位段にB01～Bnkのエンティティ、さらにその下位段にC1～cnのエンティティが構成されている。最も下のエンティティは、末端ノード（リーフ）が個々のデバイス、例えば記録再生器、再生専用器等であるとする。

【0181】ここで、リボーク処理は、各エンティティにおいて独自に実行される。例えば、最下段のエンティティC1～Cnでは、リーフのデバイスのリボーク処理が実行される。図34（b）には、最下段のエンティティの1つであるエンティティCn、3430のツリー構成を示している。エンティティCn、3430は、頂点ノード3431を持ち、末端ノードであるリーフに複数のデバイスを持つ構成である。

【0182】この末端ノードであるリーフ中に、リボーク対象となるデバイス、例えばデバイス3432があったとすると、エンティティCn、3430は、独自に更新したエンティティCn内のノードキー、リーフキーによって構成される有効化キープロック（サブEKB）を生成する。この有効化キープロックは、リボークデバイス3432においては復号できず、他のリーフを構成するデバイスにおいてのみ復号可能な暗号化キーにより構

成されるキーブロックである。エンティティC_nの管理者は、これを更新されたサブEKBとして生成する。具体的には、サブルートからリボークデバイス3432に連なるパスを構成する各ノード3431, 3434, 3435のノードキーを更新して、この更新ノードキーをリボークデバイス3432以外のリーフデバイスにおいてのみ復号可能な暗号化キーとして構成したブロックを更新サブEKBとする。この処理は、先の図3, 4において説明したリボーク処理構成において、ルートキーを、エンティティの頂点キーであるサブルートキーに置き換えた処理に対応する。

【0183】このようにエンティティC_n, 3430がリボーク処理によって更新した有効化キーブロック（サブEKB）は、上位エンティティに送信される。この場合、上位エンティティはエンティティB_nk, 3420であり、エンティティC_n, 3430の頂点ノード3431を末端ノードとして有するエンティティである。

【0184】エンティティB_nk, 3420は、下位エンティティC_n, 3430から有効化キーブロック（サブEKB）を受領すると、そのキーブロックに含まれるエンティティC_nk, 3430の頂点ノード3431に対応するエンティティB_nk, 3420の末端ノード3431を、下位エンティティC_n, 3430において更新されたキーに設定して、自身のエンティティB_nk, 3420のサブEKBの更新処理を実行する。図34

(c)にエンティティB_nk, 3420のツリー構成を示す。エンティティB_nk, 3420において、更新対象となるノードキーは、図34(c)のサブルート3421からリボークデバイスを含むエンティティを構成する末端ノード3431に至るパス上のノードキーである。すなわち、更新サブEKBを送信してきたエンティティのノード3431に連なるパスを構成する各ノード3421, 3424, 3425のノードキーが更新対象となる。これら各ノードのノードキーを更新してエンティティB_nk, 3420の新たな更新サブEKBを生成する。

【0185】さらに、エンティティB_nk, 3420が更新した有効化キーブロック（サブEKB）は、上位エンティティに送信される。この場合、上位エンティティはエンティティA_nn, 3410であり、エンティティB_nk, 3420の頂点ノード3421を末端ノードとして有するエンティティである。

【0186】エンティティA_nn, 3410は、下位エンティティB_nk, 3420から有効化キーブロック（サブEKB）を受領すると、そのキーブロックに含まれるエンティティB_nk, 3420の頂点ノード3421に対応するエンティティA_nn, 3410の末端ノード3421を、下位エンティティB_nk, 3420において更新されたキーに設定して、自身のエンティティA_nn, 3410のサブEKBの更新処理を実行する。図

34(d)にエンティティA_nn, 3410のツリー構成を示す。エンティティA_nn, 3410において、更新対象となるノードキーは、図34(d)のサブルート3411から更新サブEKBを送信してきたエンティティのノード3421に連なるパスを構成する各ノード3411, 3414, 3415のノードキーである。これら各ノードのノードキーを更新してエンティティA_nn, 3410の新たな更新サブEKBを生成する。

【0187】これらの処理を順次、上位のエンティティにおいて実行し、図29(b)で説明したルートエンティティまで実行する。この一連の処理により、デバイスのリボーク処理が完結する。なお、それぞれのエンティティにおいて更新されたサブEKBは、最終的にキー発行センター(KDC)に送信され、保管される。キー発行センター(KDC)は、すべてのエンティティの更新サブEKBに基づいて、様々なEKBを生成する。更新EKBは、リボークされたデバイスでの復号が不可能な暗号化キーブロックとなる。

【0188】デバイスのリボーク処理のシーケンス図を図35に示す。処理手順を図35のシーケンス図に従って説明する。まず、ツリー構成の最下段にあるデバイス管理エンティティ(D-E_n)は、デバイス管理エンティティ(D-E_n)内のリボーク対象のリーフを排除するために必要なキー更新を行ない、デバイス管理エンティティ(D-E_n)の新たなサブEKB(D)を生成する。更新サブEKB(D)は、上位エンティティに送付される。更新サブEKB(D)を受領した上位(親)エンティティ(P1-E_n)は、更新サブEKB(D)の更新頂点ノードに対応した末端ノードキーの更新および、その末端ノードからサブルートに至るパス上のノードキーを更新した更新サブEKB(P1)を生成する。これらの処理を順次、上位エンティティにおいて実行して、最終的に更新されたすべてのサブEKBがキー発行センター(KDC)に格納され管理される。

【0189】図36にデバイスのリボーク処理によって上位エンティティが更新処理を行なって生成する有効化キーブロック(EKB)の例を示す。

【0190】図36は、(a)に示す構成において、リボークデバイスを含む下位エンティティから更新サブEKBを受信した上位エンティティにおいて生成するEKBの例を説明する図である。リボークデバイスを含む下位エンティティの頂点ノードは、上位エンティティの末端ノード(node100)3601に対応する。

【0191】上位エンティティは、上位エンティティのサブルートから末端ノード(node100)3601までのパスに存在するノードキーを更新して新たな更新サブEKBを生成する。更新サブEKBは、図36

(b)のようになる。更新されたキーは、下線および「'」を付して示してある。このように更新された末端ノードからサブルートまでのパス上のノードキーを更

新してそのエンティティにおける更新サブEKBとする。

【0192】次に、リポーくする対象をエンティティとした場合の処理、すなわちエンティティのリポーく処理について説明する。

【0193】図37(a)は、エンティティ管理によるキー配信ツリー構造を示している。ツリー最上位にはルートノードが設定され、その数段下にエンティティA01~Ann、さらにその下位段にB01~Bnkのエンティティ、さらにその下位段にC1~cnのエンティティが構成されている。最も下のエンティティは、末端ノード(リーフ)が個々のデバイス、例えば記録再生器、再生専用器等であるとする。

【0194】ここで、リポーく処理を、エンティティCn, 3730に対して実行する場合について説明する。最下段のエンティティCn, 3730は、図37(b)に示すように頂点ノード3431を持ち、末端ノードであるリーフに複数のデバイスを持つ構成である。

【0195】エンティティCn, 3730をリポーくすることにより、エンティティCn, 3730に属するすべてのデバイスのツリー構造からの一括排除が可能となる。エンティティCn, 3730のリポーく処理は、エンティティCn, 3730の上位エンティティであるエンティティBnk, 3720において実行される。エンティティBnk, 3720は、エンティティCn, 3730の頂点ノード3731を末端ノードとして有するエンティティである。

【0196】エンティティBnk, 3720は、下位エンティティCn, 3730のリポーくを実行する場合、エンティティCnk, 3730の頂点ノード3731に対応するエンティティBnk, 3720の末端ノード3731を更新し、さらに、そのリポーくエンティティ3730からエンティティBnk, 3720のサブルートまでのパス上のノードキーの更新を行ない有効化キープロックを生成して更新サブEKBを生成する。更新対象となるノードキーは、図37(c)のサブルート3721からリポーくエンティティの頂点ノードを構成する末端ノード3731に至るパス上のノードキーである。すなわち、ノード3721, 3724, 3725, 3731のノードキーが更新対象となる。これら各ノードのノードキーを更新してエンティティBnk, 3720の新たな更新サブEKBを生成する。

【0197】あるいは、エンティティBnk, 3720は、下位エンティティCn, 3730のリポーくを実行する場合、エンティティCnk, 3730の頂点ノード3731に対応するエンティティBnk, 3720の末端ノード3731は更新せず、そのリポーくエンティティ3730からエンティティBnk, 3720のサブルートまでのパス上の末端ノード3731を除くノードキーの更新を行ない有効化キープロックを生成して更新サ

ブEKBを生成してもよい。

【0198】さらに、エンティティBnk, 3720が更新した有効化キープロック(サブEKB)は、上位エンティティに送信される。この場合、上位エンティティはエンティティAnn, 3710であり、エンティティBnk, 3720の頂点ノード3721を末端ノードとして有するエンティティである。

【0199】エンティティAnn, 3710は、下位エンティティBnk, 3720から有効化キープロック(サブEKB)を受領すると、そのキープロックに含まれるエンティティBnk, 3720の頂点ノード3721に対応するエンティティAnn, 3710の末端ノード3721を、下位エンティティBnk, 3720において更新されたキーに設定して、自身のエンティティAnn, 3710のサブEKBの更新処理を実行する。図37(d)にエンティティAnn, 3710のツリー構成を示す。エンティティAnn, 3710において、更新対象となるノードキーは、図37(d)のサブルート3711から更新サブEKBを送信してきたエンティティのノード3721に連なるパスを構成する各ノード3711, 3714, 3715のノードキーである。これら各ノードのノードキーを更新してエンティティAnn, 3710の新たな更新サブEKBを生成する。

【0200】これらの処理を順次、上位のエンティティにおいて実行し、図29(b)で説明したルートエンティティまで実行する。この一連の処理により、エンティティのリポーく処理が完結する。なお、それぞれのエンティティにおいて更新されたサブEKBは、最終的にキー発行センター(KDC)に送信され、保管される。キー発行センター(KDC)は、すべてのエンティティの更新サブEKBに基づいて、様々なEKBを生成する。更新EKBは、リポーくされたエンティティに属するデバイスでの復号が不可能な暗号化キープロックとなる。

【0201】エンティティのリポーく処理のシーケンス図を図38に示す。処理手順を図38のシーケンス図に従って説明する。まず、エンティティをリポーくしようとするエンティティ管理エンティティ(E-En)は、エンティティ管理エンティティ(E-En)内のリポーく対象の末端ノードを排除するために必要なキー更新を行ない、エンティティ管理エンティティ(E-En)の新たなサブEKB(E)を生成する。更新サブEKB

(E)は、上位エンティティに送付される。更新サブEKB(E)を受領した上位(親)エンティティ(P1-En)は、更新サブEKB(E)の更新頂点ノードに対応した末端ノードキーの更新および、その末端ノードからサブルートに至るパス上のノードキーを更新した更新サブEKB(P1)を生成する。これらの処理を順次、上位エンティティにおいて実行して、最終的に更新されたすべてのサブEKBがキー発行センター(KDC)に格納され管理される。キー発行センター(KDC)は、

すべてのエンティティの更新サブEKBに基づいて、様々なEKBを生成する。更新EKBは、リボークされたエンティティに属するデバイスでの復号が不可能な暗号化キーブロックとなる。

【0202】図39にリボークされた下位エンティティと、リボークを行なった上位エンティティの対応を説明する図を示す。上位エンティティの末端ノード3901は、エンティティのリボークにより更新され、上位エンティティのツリーにおける末端ノード3901からサブルートまでのパスに存在するノードキーの更新により、新たなサブEKBが生成される。その結果、リボークされた下位エンティティの頂点ノード3902のノードキーと、上位エンティティの末端ノード3901のノードキーは不一致となる。エンティティのリボーク後にキー発行センター(KDC)によって生成されるEKBは、上位エンティティにおいて更新された末端ノード3901のキーに基づいて生成されることになるので、その更新キーを保有しない下位エンティティのリーフに対応するデバイスは、キー発行センター(KDC)によって生成されるEKBの復号が不可能になる。

【0203】なお、上述の説明では、デバイスを管理する最下段のエンティティのリボーク処理について説明したが、ツリーの中段にあるエンティティ管理エンティティをその上位エンティティがリボークする処理も上記と同様のプロセスによって可能である。中段のエンティティ管理エンティティをリボークすることにより、リボークされたエンティティ管理エンティティの下位に属するすべての複数エンティティおよびデバイスを一括してリボーク可能となる。

【0204】このように、エンティティ単位でのリボークを実行することにより、1つ1つのデバイス単位で実行するリボーク処理に比較して簡易なプロセスでのリボーク処理が可能となる。

【0205】[エンティティのケイパビリティ管理] 次に、エンティティ単位でのキー配信ツリー構成において、各エンティティの許容するケイパビリティ(Capability)を管理して、ケイパビリティに応じたコンテンツ配信を行なう処理構成について説明する。ここでケイパビリティとは、例えば特定の圧縮音声データの復号が可能であるとか、特定の音声再生方式を許容するとか、あるいは特定の画像処理プログラムを処理できる等、デバイスがどのようなコンテンツ、あるいはプログラム等を処理できるデバイスであるか、すなわちデバイスのデータ処理能力の定義情報である。

【0206】図40にケイパビリティを定義したエンティティ構成例を示す。キー配信ツリー構成の最頂点にルートノードが位置し、下層に複数のエンティティが接続されて各ノードが2分岐を持つツリー構成である。ここで、例えばエンティティ4001は、音声再生方式A、B、Cのいずれかを許容するケイパビリティを持つ

エンティティとして定義される。具体的には、例えばある音声圧縮プログラムA、B、またはC方式で圧縮した音楽データを配信した場合に、エンティティ4001以下に構成されたエンティティに属するデバイスは圧縮データを伸長する処理が可能である。

【0207】同様にエンティティ4002は音声再生方式BまたはC、エンティティ4003は音声再生方式AまたはB、エンティティ4004は音声再生方式B、エンティティ4005は音声再生方式Cを処理することが可能なケイパビリティを持つエンティティとして定義される。

【0208】一方、エンティティ4021は、画像再生方式p、q、rを許容するエンティティとして定義され、エンティティ4022は方式p、qの画像再生方式、エンティティ4023は方式pの画像再生が可能なケイパビリティを持つエンティティとして定義される。

【0209】このような各エンティティのケイパビリティ情報は、キー発行センター(KDC)において管理される。キー発行センター(KDC)は、例えばあるコンテンツプロバイダが特定の圧縮プログラムで圧縮した音楽データを様々なデバイスに配信したい場合、その特定の圧縮プログラムを再生可能なデバイスに対してのみ復号可能な有効化キーブロック(EKB)を各エンティティのケイパビリティ情報に基づいて生成することができる。コンテンツを提供するコンテンツプロバイダは、ケイパビリティ情報に基づいて生成した有効化キーブロック(EKB)によって暗号化したコンテンツキーを配信し、そのコンテンツキーで暗号化した圧縮音声データを各デバイスに提供する。この構成により、データの処理が可能なデバイスに対してのみ特定の処理プログラムを確実に提供することが可能となる。

【0210】なお、図40では全てのエンティティについてケイパビリティ情報を定義している構成であるが、図40の構成のようにすべてのエンティティにケイパビリティ情報を定義することは必ずしも必要ではなく、例えば図41に示すようにデバイスが属する最下段のエンティティについてのみケイパビリティを定義して、最下段のエンティティに属するデバイスのケイパビリティをキー発行センター(KDC)において管理して、コンテンツプロバイダが望む処理の可能なデバイスにのみ復号可能な有効化キーブロック(EKB)を最下段のエンティティに定義されたケイパビリティ情報に基づいて生成する構成としてもよい。図41では、末端ノードにデバイスが定義されたエンティティ4101=4105におけるケイパビリティが定義され、これらのエンティティについてのケイパビリティをキー発行センター(KDC)において管理する構成である。例えばエンティティ4101には音声再生については方式B、画像再生については方式rの処理が可能なデバイスが属している。エンティティ4102には音声再生については方式A、画像再

生については方式 q の処理が可能なデバイスが属している等である。

【0211】図42にキー発行センター（KDC）において管理するケイパビリティ管理テーブルの構成例を示す。ケイパビリティ管理テーブルは、図42（a）のようなデータ構成を持つ。すなわち、各エンティティを識別する識別子としてのエンティティID、そのエンティティに定義されたケイパビリティを示すケイパビリティリスト、このケイパビリティリストは図42（b）に示すように、例えば音声データ再生処理方式（A）が処理可能であれば「1」、処理不可能であれば「0」、音声データ再生処理方式（B）が処理可能であれば「1」、処理不可能であれば「0」…等、様々な態様のデータ処理についての可否を1ビットづつ「1」または「0」を設定して構成されている。なお、このケイパビリティ情報の設定方法はこのような形式に限らず、エンティティの管理デバイスについてのケイパビリティを識別可能であれば他の構成でもよい。

【0212】ケイパビリティ管理テーブルには、さらに、各エンティティのサブEKB、あるいはサブEKBが別のデータベースに格納されている場合は、サブEKBの識別情報が格納され、さらに、各エンティティのサブルートノード識別データが格納される。

【0213】キー発行センター（KDC）は、ケイパビリティ管理テーブルに基づいて、例えば特定のコンテンツの再生可能なデバイスのみが復号可能な有効化キープロック（EKB）を生成する。図43を用いて、ケイパビリティ情報に基づく有効化キープロックの生成処理について説明する。

【0214】まず、ステップS4301において、キー発行センター（KDC）は、ケイパビリティ管理テーブルから、指定されたケイパビリティを持つエンティティを選択する。具体的には、例えばコンテンツプロバイダが音声データ再生処理方式Aに基づく再生可能なデータを配信したい場合は、図42（a）のケイパビリティリストから、例えば音声データ再生処理（方式A）の項目が「1」に設定されたエンティティを選択する。

【0215】次に、ステップS4302において、選択されたエンティティによって構成される選択エンティティIDのリストを生成する。次に、ステップS4303で、選択エンティティIDによって構成されるツリーに必要なパス（キー配信ツリー構成のパス）を選択する。ステップS4304では、選択エンティティIDのリストに含まれる全てのパス選択が完了したか否かを判定し、完了するまで、ステップS4303においてパスを生成する。これは、複数のエンティティが選択された場合に、それぞれのパスを順次選択する処理を意味している。

【0216】選択エンティティIDのリストに含まれる全てのパス選択が完了すると、ステップS4305に進

み、選択したパスと、選択エンティティによってのみ構成されるキー配信ツリー構造を構築する。

【0217】次に、ステップS4306において、ステップS4305で生成したツリー構造のノードキーの更新処理を行ない、更新ノードキーを生成する。さらに、ツリーを構成する選択エンティティのサブEKBをケイパビリティ管理テーブルから取り出し、サブEKBと、ステップS4306で生成した更新ノードキーとに基づいて選択エンティティのデバイスにおいてのみ復号可能な有効化キープロック（EKB）を生成する。このようにして生成した有効化キープロック（EKB）は、特定のケイパビリティを持つデバイスにおいてのみ利用、すなわち復号可能な有効化キープロック（EKB）となる。この有効化キープロック（EKB）で例えばコンテンツキーを暗号化して、そのコンテンツキーで特定プログラムに基づいて圧縮したコンテンツを暗号化してデバイスに提供することで、キー発行センター（KDC）によって選択された特定の処理可能なデバイスにおいてのみコンテンツが利用される。

【0218】このようにキー発行センター（KDC）は、ケイパビリティ管理テーブルに基づいて、例えば特定のコンテンツの再生可能なデバイスのみが復号可能な有効化キープロック（EKB）を生成する。従って、新たなエンティティが登録される場合には、その新規登録エンティティのケイパビリティを予め取得することが必要となる。このエンティティ新規登録に伴うケイパビリティ通知処理について図44を用いて説明する。

【0219】図44は、新規エンティティがキー配信ツリー構成に参加する場合のケイパビリティ通知処理シーケンスを示した図である。

【0220】新たにツリー構成中に追加される新規（子）エンティティ（N-E n）は、上位（親）エンティティ（P-E n）に対して新規登録要求を実行する。なお、各エンティティは、公開鍵暗号方式に従った公開鍵を保有し、新規エンティティは自己の公開鍵を登録要求に際して上位エンティティ（P-E n）に送付する。

【0221】登録要求を受領した上位エンティティ（P-E n）は、受領した新規（子）エンティティ（N-E n）の公開鍵を証明書発行局（CA：Certificate Authority）に転送し、CAの署名を付加した新規（子）エンティティ（N-E n）の公開鍵を受領する。これらの手続きは、上位エンティティ（P-E n）と新規（子）エンティティ（N-E n）との相互認証の手続きとして行われる。

【0222】これらの処理により、新規登録要求エンティティの認証が終了すると、上位エンティティ（P-E n）は、新規（子）エンティティ（N-E n）の登録を許可し、新規（子）エンティティ（N-E n）のノードキーを新規（子）エンティティ（N-E n）に送信する。このノードキーは、上位エンティティ（P-E n）

の末端ノードの1つのノードキーであり、かつ、新規(子)エンティティ(N-E_n)の頂点ノード、すなわちサブルートキーに対応する。

【0223】このノードキー送信が終了すると、新規(子)エンティティ(N-E_n)は、新規(子)エンティティ(N-E_n)のツリー構成を構築し、構築したツリーの頂点に受信した頂点ノードのサブルートキーを設定し、各ノード、リーフのキーを設定して、エンティティ内の有効化キーブロック(サブEKB)を生成する。一方、上位エンティティ(P-E_n)も、新規(子)エンティティ(N-E_n)の追加により、有効化する末端ノードを追加した上位エンティティ(P-E_n)内のサブEKBを生成する。

【0224】新規(子)エンティティ(N-E_n)は、新規(子)エンティティ(N-E_n)内のノードキー、リーフキーによって構成されるサブEKBを生成すると、これを上位エンティティ(P-E_n)に送信し、さらに、自己のエンティティで管理するデバイスについてのケイパビリティ情報を上位エンティティに通知する。

【0225】新規(子)エンティティ(N-E_n)からサブEKBおよびケイパビリティ情報を受信した上位エンティティ(P-E_n)は、受信したサブEKBとケイパビリティ情報と、上位エンティティ(P-E_n)の更新したサブEKBとをキー発行センター(KDC: Key Distribute Center)に送信する。

【0226】キー発行センター(KDC)は、受領したエンティティのサブEKBおよびケイパビリティ情報とを図42で説明したケイパビリティ管理テーブルに登録し、ケイパビリティ管理テーブルを更新する。キー発行センター(KDC)は、更新したケイパビリティ管理テーブルに基づいて、様々な態様のEKB、すなわち特定のケイパビリティを持つエンティティあるいはデバイスのみが復号可能なEKBを生成することが可能となる。

【0227】以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が該実施例の修正や代用を成し得ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参照すべきである。

【0228】

【発明の効果】以上、説明したように、本発明の情報処理システムおよび方法によれば、複数のデバイスをリーフとして構成したツリーのルートからリーフまでのパス上のルート、ノード、およびリーフに各々キーを対応付けたキーツリーをデバイスのデータ処理能力としてのケイパビリティに基づいて区分したサブツリーを設定し、それぞれのサブツリーの管理主体であるエンティティにおいて、エンティティ内で有効なサブ有効化キーブロック(サブEKB)を生成するとともに、キー発行センタ

ー(KDC)において、エンティティのケイパビリティ情報を管理して、共通のケイパビリティを持つエンティティにおいてのみ復号可能な有効化キーブロック(EKB)を生成する構成としたので、特定のデバイスにおいてのみ処理可能なデータを、そのデバイスにおいてのみ復号可能なデータとして提供することが可能となる。

【0229】さらに、本発明の情報処理システムおよび方法によれば、キー発行センター(KDC)は、複数のエンティティ各々の識別子、ケイパビリティ情報、サブ有効化キーブロック(サブEKB)情報とを対応付けたケイパビリティ管理テーブルに基づいて、デバイスに対する配信データの処理可能なエンティティを選択して、様々なケイパビリティに応じた様々なEKBの生成が可能となる。

【0230】さらに、本発明の情報処理システムおよび方法によれば、エンティティでのデバイスあるいはエンティティのリボーク処理が実行可能であり、一括したデバイス管理の場合のデバイス増大にともなう処理量の増加が防止される。

【0231】さらに、本発明の情報処理システムおよび方法によれば、各エンティティの末端ノードにリザーブノードを設定する構成としたので、管理デバイスまたは管理エンティティの増加にも対応可能となる。

【図面の簡単な説明】

【図1】本発明の情報処理システムの構成例を説明する図である。

【図2】本発明の情報処理システムにおいて適用可能な記録再生装置の構成例を示すブロック図である。

【図3】本発明の情報処理システムにおける各種キー、データの暗号化処理について説明するツリー構成図である。

【図4】本発明の情報処理システムにおける各種キー、データの配布に使用される有効化キーブロック(EKB)の例を示す図である。

【図5】本発明の情報処理システムにおけるコンテンツキーの有効化キーブロック(EKB)を使用した配布例と復号処理例を示す図である。

【図6】本発明の情報処理システムにおける有効化キーブロック(EKB)のフォーマット例を示す図である。

【図7】本発明の情報処理システムにおける有効化キーブロック(EKB)のタグの構成を説明する図である。

【図8】本発明の情報処理システムにおける有効化キーブロック(EKB)と、コンテンツキー、コンテンツを併せて配信するデータ構成例を示す図である。

【図9】本発明の情報処理システムにおける有効化キーブロック(EKB)と、コンテンツキー、コンテンツを併せて配信した場合のデバイスでの処理例を示す図である。

【図10】本発明の情報処理システムにおける有効化キーブロック(EKB)とコンテンツを記録媒体に格納し

た場合の対応について説明する図である。

【図 1 1】本発明の情報処理システムにおける有効化キーブロック（E K B）と、コンテンツキーを送付する処理を従来の送付処理と比較した図である。

【図 1 2】本発明の情報処理システムにおいて適用可能な共通鍵暗号方式による認証処理シーケンスを示す図である。

【図 1 3】本発明の情報処理システムにおける有効化キーブロック（E K B）と、認証キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その 1）である。

【図 1 4】本発明の情報処理システムにおける有効化キーブロック（E K B）と、認証キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その 2）である。

【図 1 5】本発明の情報処理システムにおいて適用可能な公開鍵暗号方式による認証処理シーケンスを示す図である。

【図 1 6】本発明の情報処理システムにおいて公開鍵暗号方式による認証処理を用いて有効化キーブロック（E K B）と、コンテンツキーを併せて配信する処理を示す図である。

【図 1 7】本発明の情報処理システムにおいて有効化キーブロック（E K B）と、暗号化プログラムデータを併せて配信する処理を示す図である。

【図 1 8】本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（I C V）の生成に使用する M A C 値生成例を示す図である。

【図 1 9】本発明の情報処理システムにおける有効化キーブロック（E K B）と、I C V 生成キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その 1）である。

【図 2 0】本発明の情報処理システムにおける有効化キーブロック（E K B）と、I C V 生成キーを併せて配信するデータ構成と、デバイスでの処理例を示す図（その 2）である。

【図 2 1】本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（I C V）をメディアに格納した場合のコピー防止機能を説明する図である。

【図 2 2】本発明の情報処理システムにおいて適用可能なコンテンツ・インテグリティ・チェック値（I C V）をコンテンツ格納媒体と別に管理する構成を説明する図である。

【図 2 3】本発明の情報処理システムにおける階層ツリー構造のカテゴリ分類の例を説明する図である。

【図 2 4】本発明の情報処理システムにおける簡略化有効化キーブロック（E K B）の生成過程を説明する図である。

【図 2 5】本発明の情報処理システムにおける有効化キ

ーブロック（E K B）の生成過程を説明する図である。

【図 2 6】本発明の情報処理システムにおける簡略化有効化キーブロック（E K B）を説明する図である。

【図 2 7】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成について説明する図である。

【図 2 8】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成の詳細について説明する図である。

【図 2 9】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成について説明する図である。

【図 3 0】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのリザーブノードについて説明する図である。

【図 3 1】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成での新規エンティティ登録処理シーケンスについて説明する図である。

【図 3 2】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成での新規エンティティと上位エンティティの関係について説明する図である。

【図 3 3】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成で用いるサブ E K B について説明する図である。

【図 3 4】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのデバイスリボーク処理について説明する図である。

【図 3 5】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのデバイスリボーク処理シーケンスについて説明する図である。

【図 3 6】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのデバイスリボーク時の更新サブ E K B について説明する図である。

【図 3 7】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのエンティティリボーク処理について説明する図である。

【図 3 8】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのエンティティリボーク処理シーケンスについて説明する図である。

【図 3 9】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのリボークエンティティと上位エンティティの関係について説明する図である。

【図 4 0】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのケイパビリティ設定について説明する図である。

【図 4 1】本発明の情報処理システムにおける階層ツリー構造のエンティティ管理構成でのケイパビリティ設定について説明する図である。

【図 4 2】本発明の情報処理システムにおけるキー発行

センター (KDC) の管理するケイパビリティ管理テーブル構成を説明する図である。

【図 4 3】本発明の情報処理システムにおけるキー発行センター (KDC) の管理するケイパビリティ管理テーブルに基づく EKB 生成処理フロー図である。

【図 4 4】本発明の情報処理システムにおける新規エンティティ登録時のケイパビリティ通知処理を説明する図である。

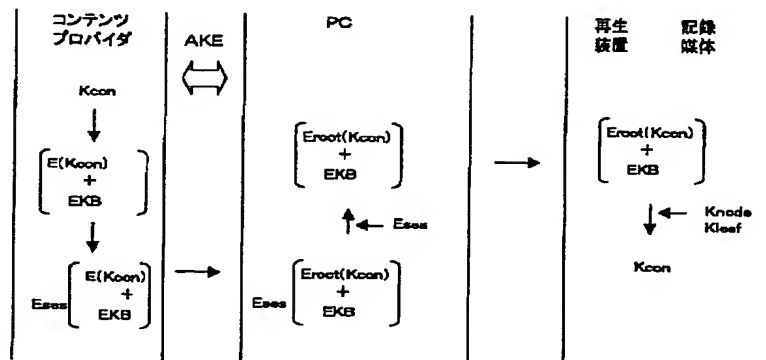
【符号の説明】

10 コンテンツ配信側
11 インターネット
12 衛星放送
13 電話回線
14 メディア
20 コンテンツ受信側
21 パーソナルコンピュータ (PC)
22 ポータブルデバイス (PD)
23 携帯電話、PDA
24 記録再生器
25 再生専用器
30 メディア
100 記録再生装置
110 バス
120 入出力 I/F
130 MPEGコーデック
140 入出力 I/F
141 A/D, D/Aコンバータ
150 暗号処理手段
160 ROM
170 CPU
180 メモリ
190 ドライブ
195 記録媒体
601 バージョン
602 デブス
603 データポインタ

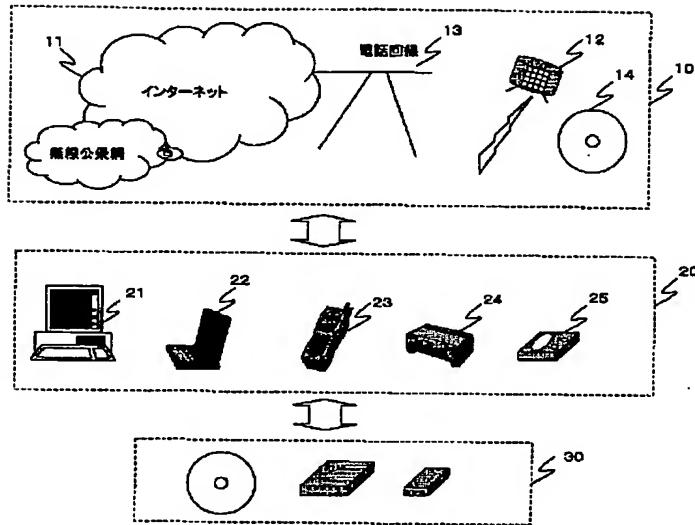
* 604 タグポインタ
605 署名ポインタ
606 データ部
607 タグ部
608 署名
1101 記録デバイス
2301 ルートキー
2302 ノードキー
2303 リーフキー
10 2304 カテゴリノード
2306 サブカテゴリノード
2701 エンティティ
2702 サブルート
2811, 2851 サブルート
2812, 2852 エンティティ末端ノード
2901, 2902 エンティティ
2950 リザーブノード
2970 管理末端ノード
3011, 3012, 3013 エンティティ
20 3021, 3022, 3023 リザーブノード
3201 末端ノード
3202 頂点ノード
3301, 3302 末端ノード
3303 新規エンティティ追加末端ノード
3410, 3420, 3430 エンティティ
3411, 3421, 3431 サブルート
3432 リボークデバイスノード
3601 末端ノード
3710, 3720, 3730 エンティティ
30 3711, 3721, 3731 サブルート
3901 末端ノード
3902 頂点ノード (サブルートノード)
4001~4005 エンティティ
4021, 4022, 4023 エンティティ
4101~4105 エンティティ

*

【図 16】



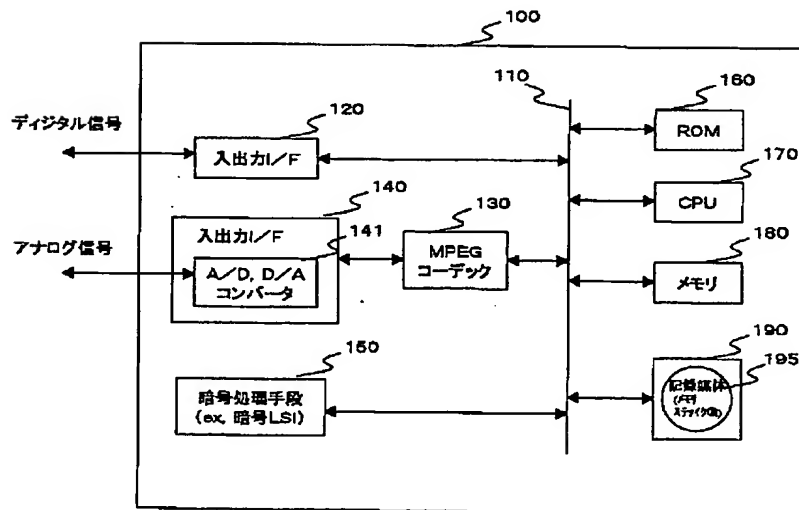
【図1】



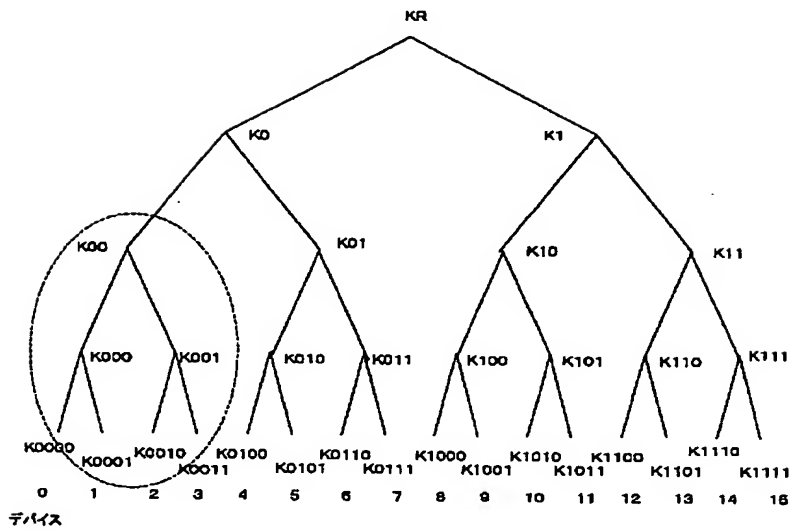
【図10】



【図2】



【図3】



【図4】

(A) 有効化キーブロック(EKB: Enabling Key Block) 例1

デバイス0, 1, 2にバージョン:tのノードキーを送付

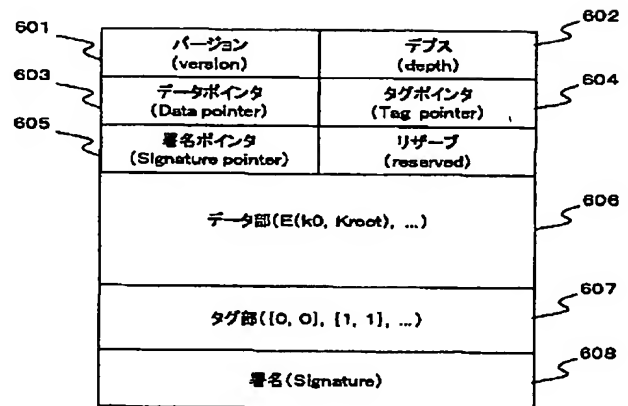
| バージョン(Version):t | |
|------------------|----------------------|
| インデックス | 暗号化キー |
| 0 | Enc(K(t)0, K(t)R) |
| 00 | Enc(K(t)00, K(t)0) |
| 000 | Enc(K000, K(t)00) |
| 001 | Enc(K(t)001, K(t)00) |
| 0010 | Enc(K0010, K(t)001) |

(B) 有効化キーブロック(EKB: Enabling Key Block) 例2

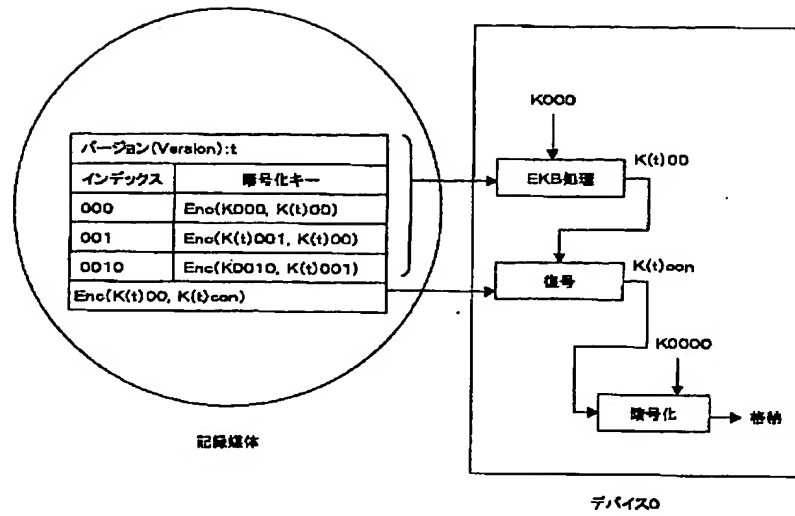
デバイス0, 1, 2にバージョン:tのノードキーを送付

| バージョン(Version):t | |
|------------------|----------------------|
| インデックス | 暗号化キー |
| 000 | Enc(K000, K(t)00) |
| 001 | Enc(K(t)001, K(t)00) |
| 0010 | Enc(K0010, K(t)001) |

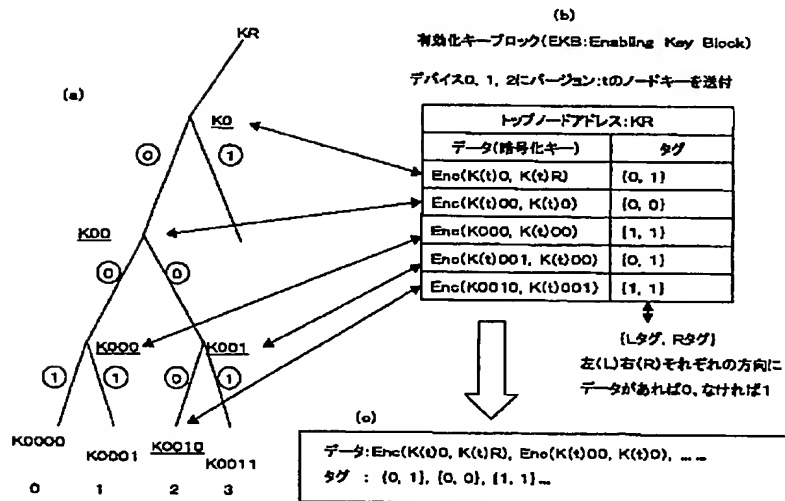
【図6】



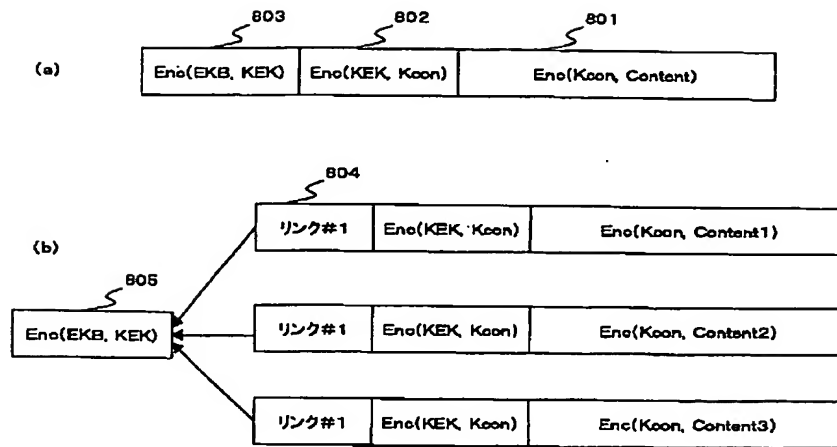
【図5】



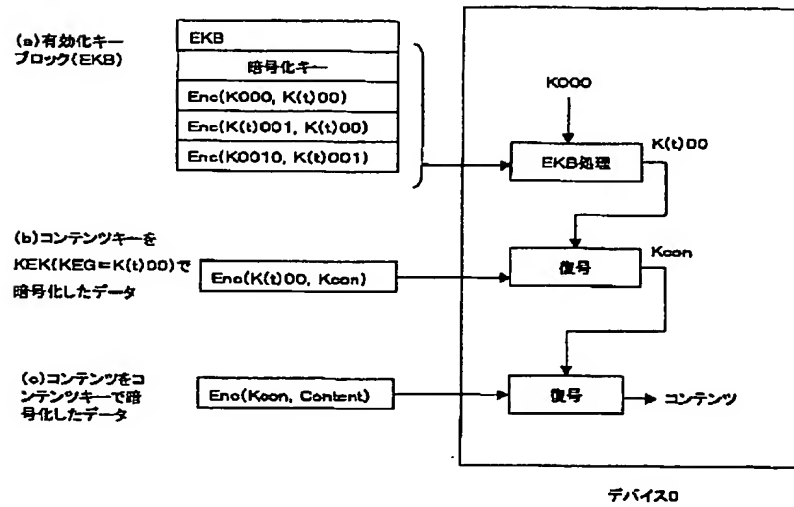
【図7】



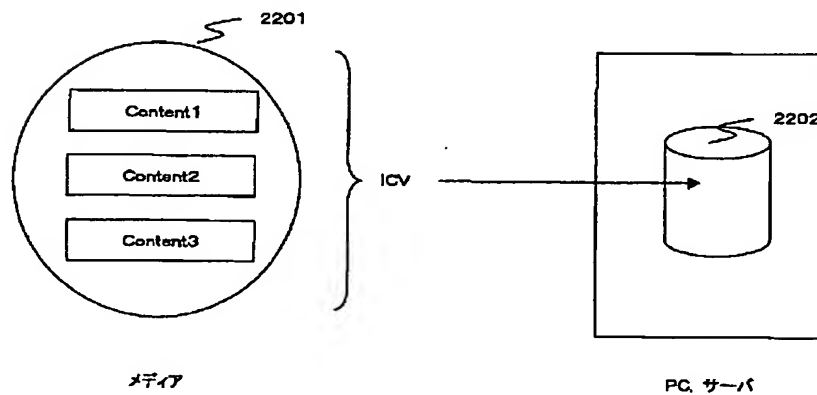
【図8】



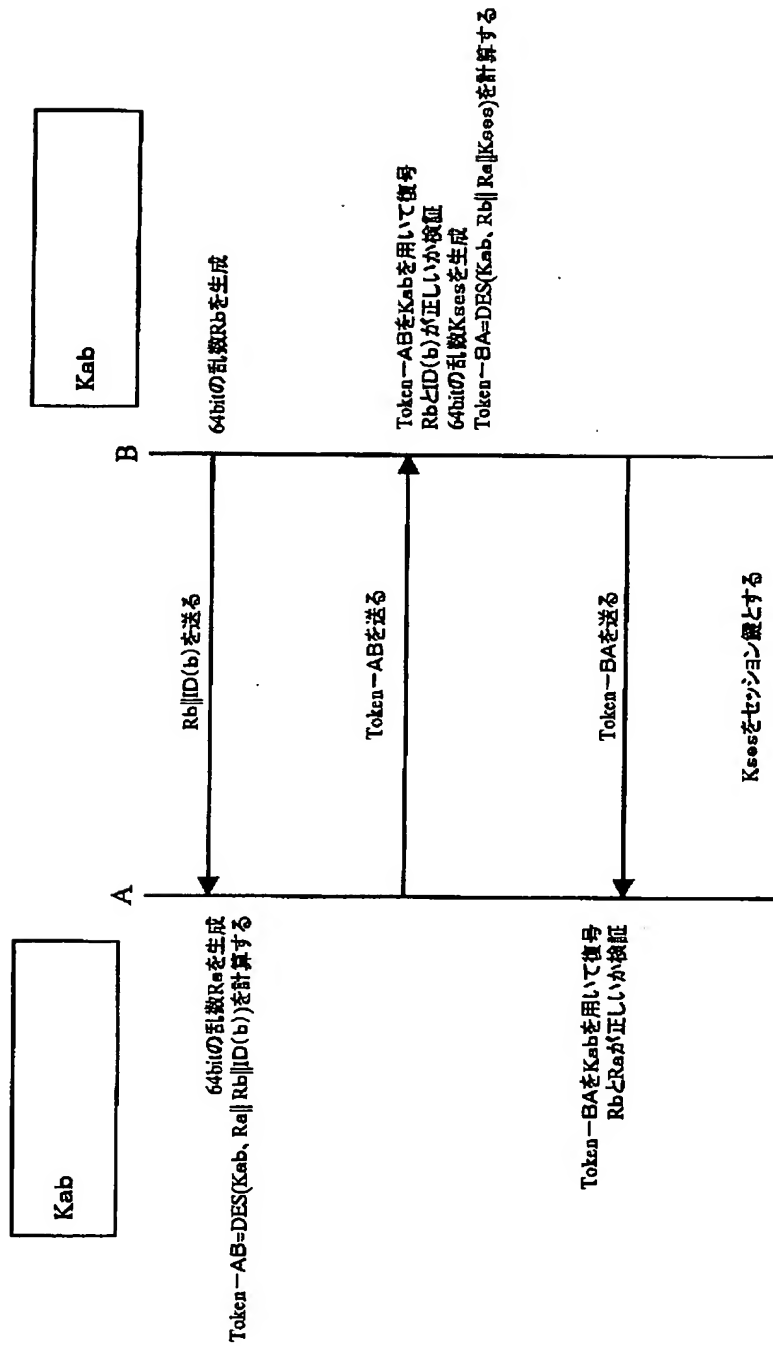
【図9】



【図22】

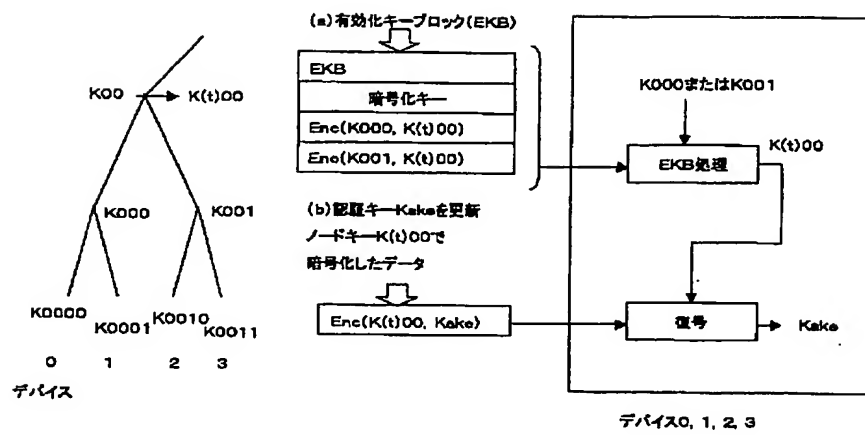


【図12】

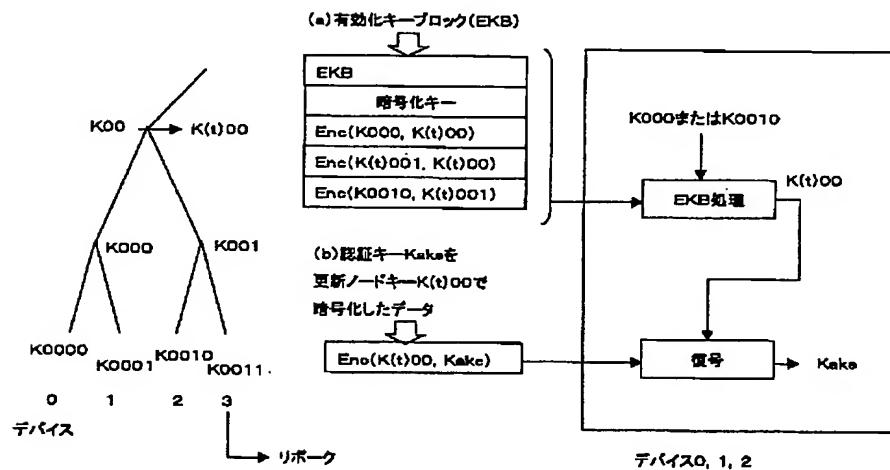


ISO/IEC 9798-2 対称鍵暗号技術を用いた相互認証および鍵共有方式

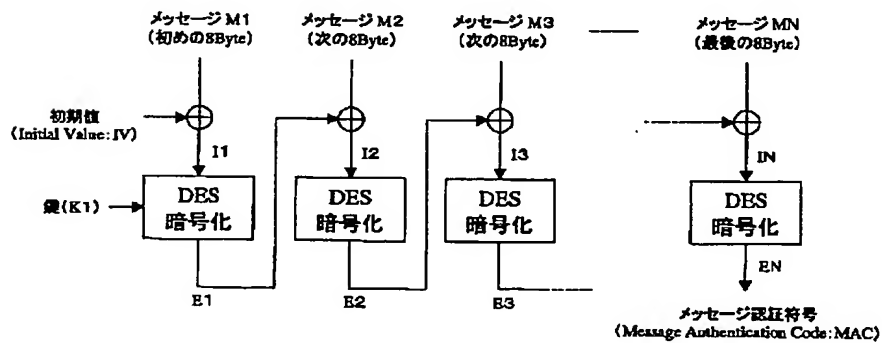
【図13】



【図14】

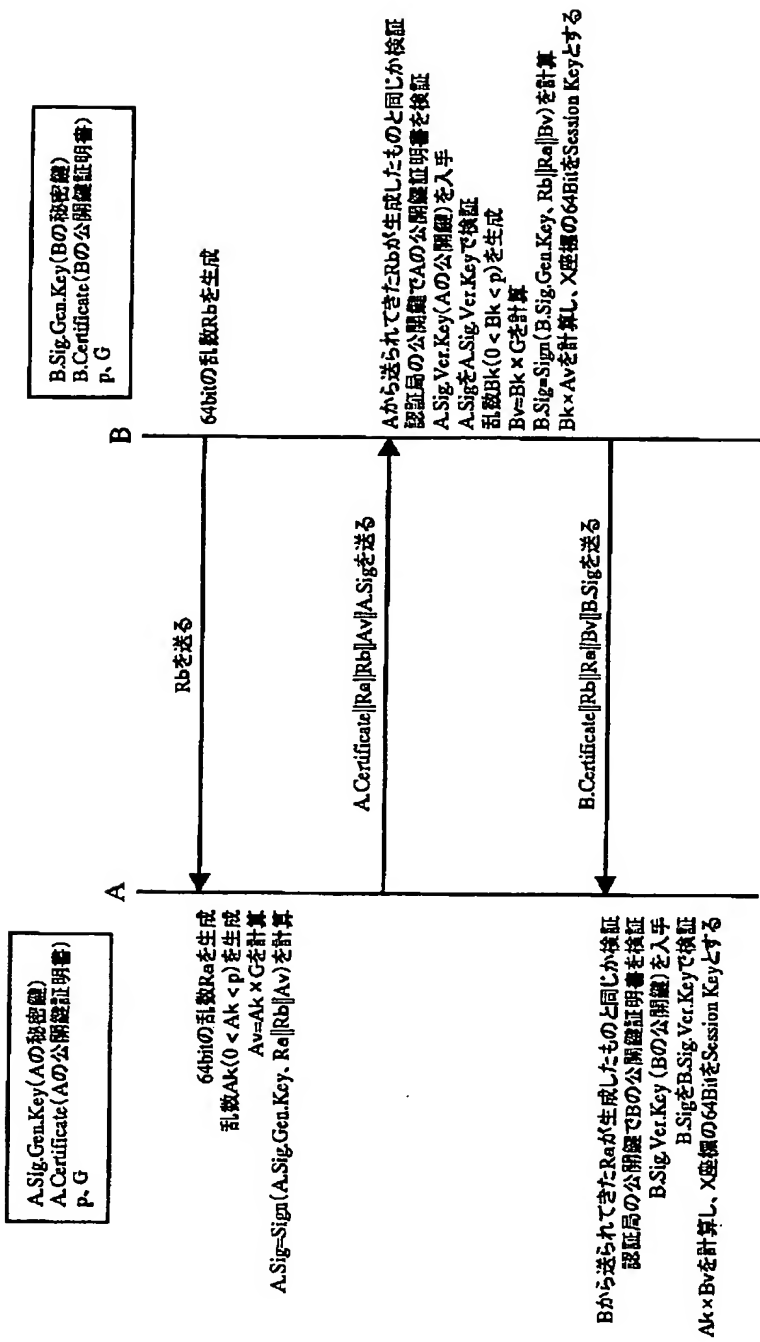


【図18】



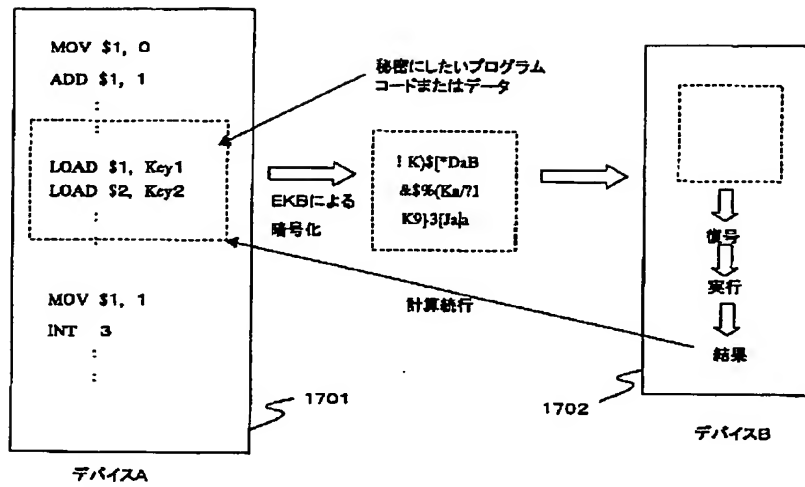
⊕ : 排他的論理和処理(8バイト単位)

【図15】

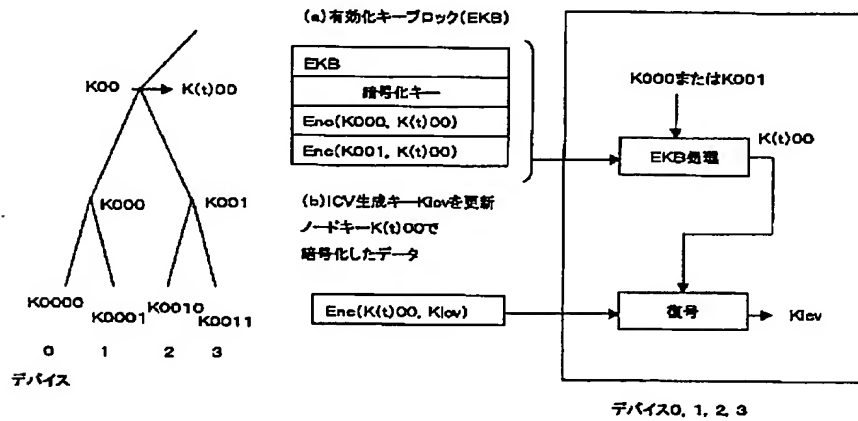


ISO/IEC 9798-3 非対称鍵暗号技術を用いた相互認証および鍵共有方式

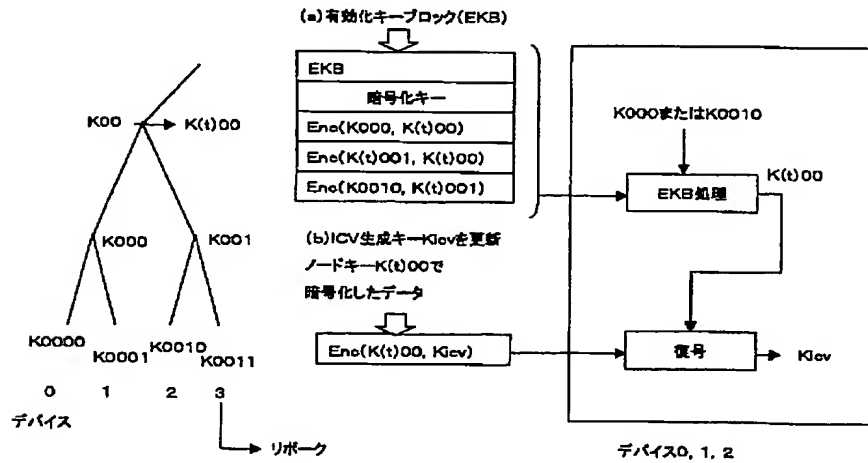
【図17】



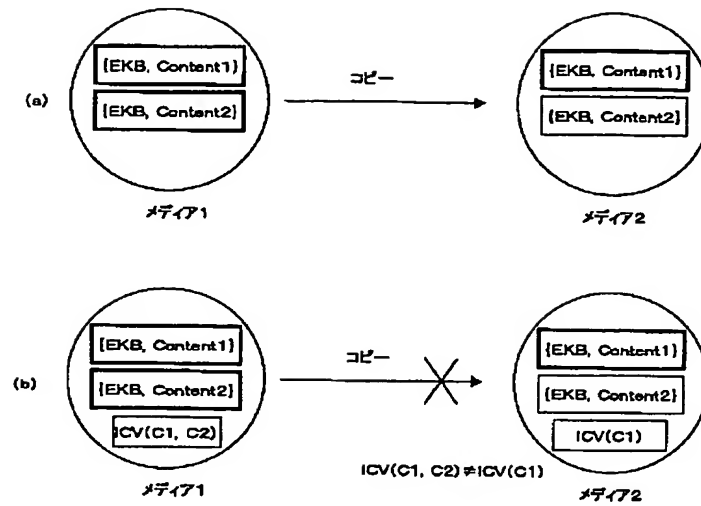
【図19】



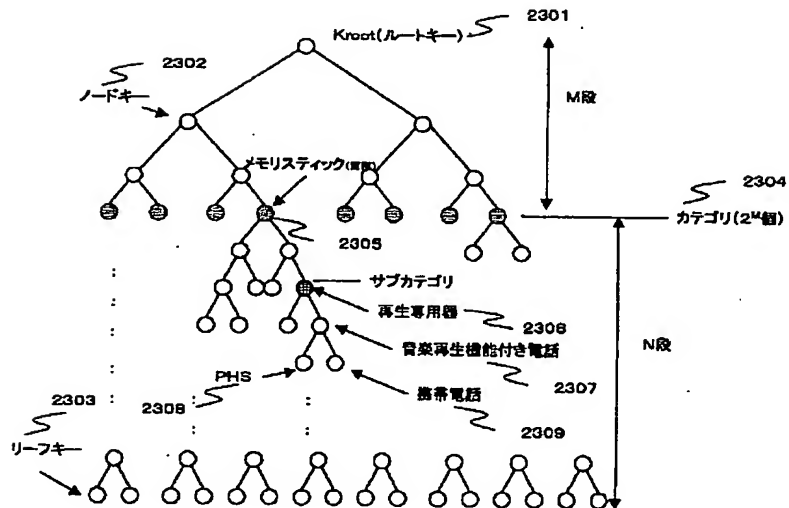
【図20】



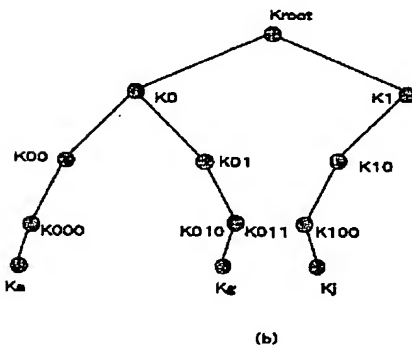
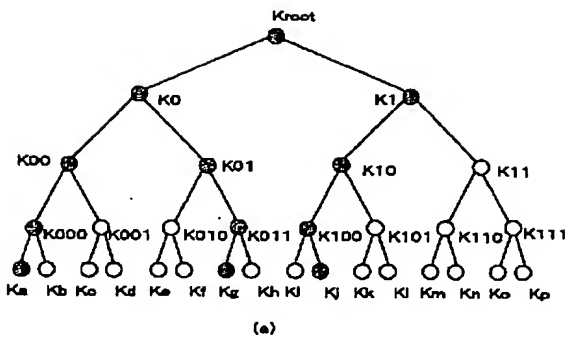
【図21】



【図23】

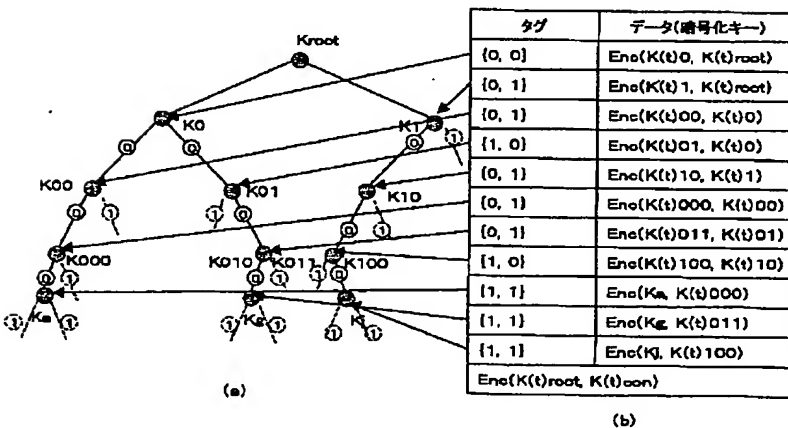


【図24】



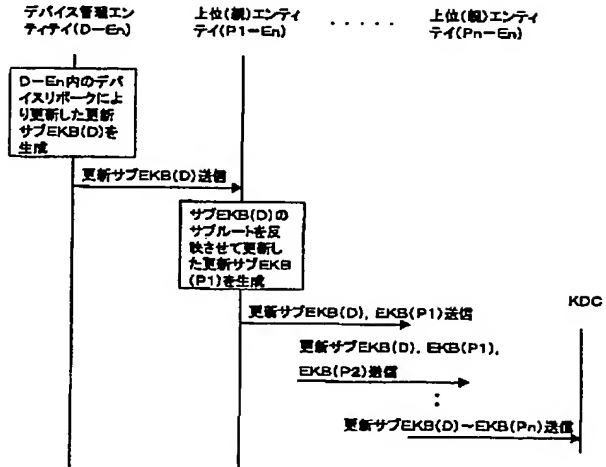
【図25】

有効化キープロック(EKB:Enabling Key Block)を用いた
デバイスKa, Kc, Kjへのバージョンtのコンテンツキー送付処理

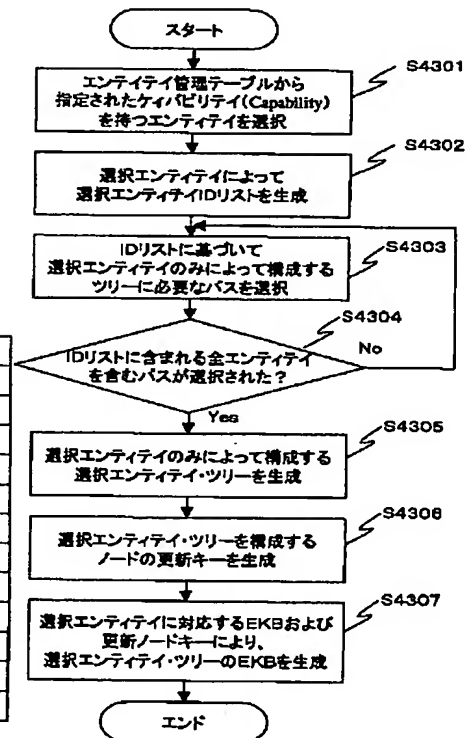


【図35】

デバイスのリポーク処理

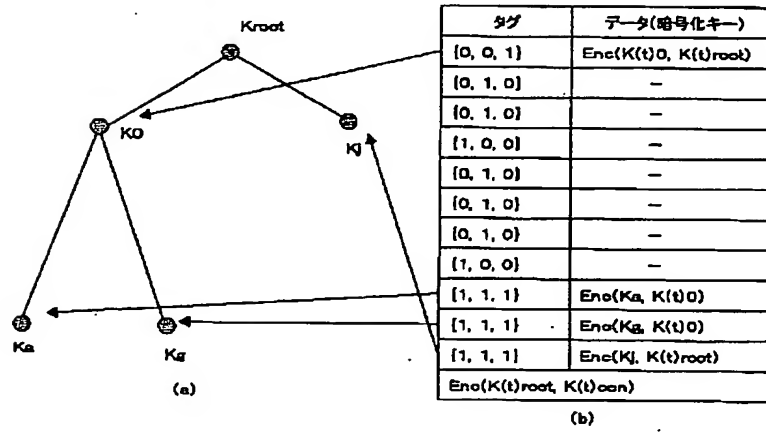


【図43】

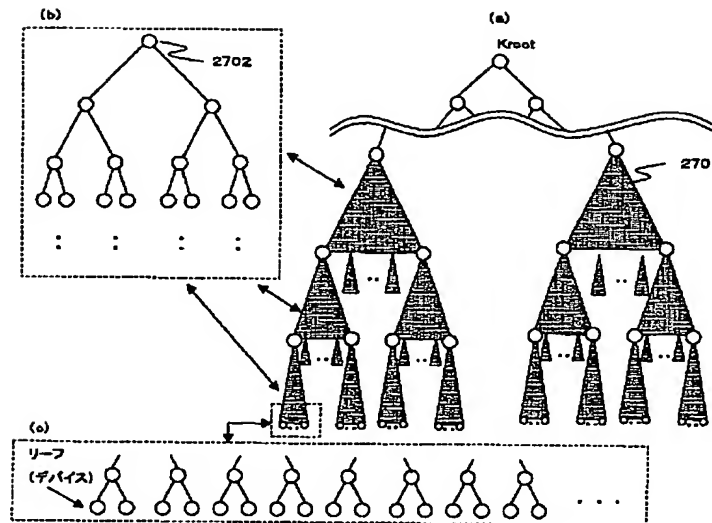


【図26】

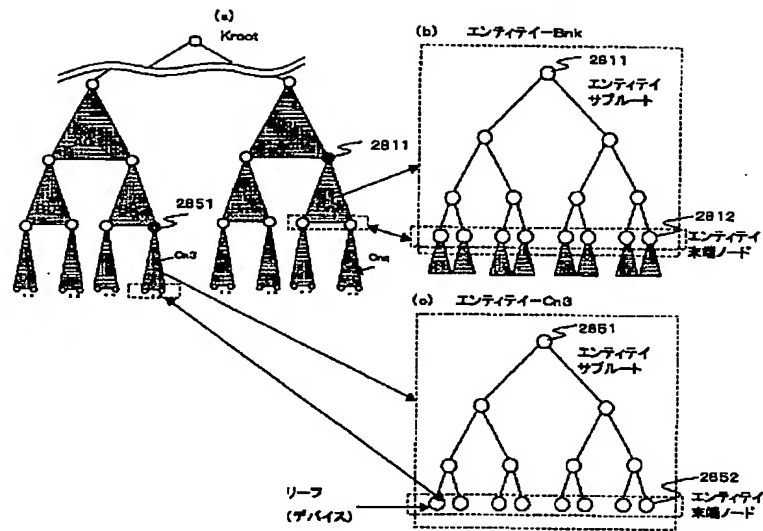
階層化した有効化キーブロック(EKB: Enabling Key Block)を用いた
デバイスKa, Kg, Kjへのバージョンtのコンテンツキー送付処理



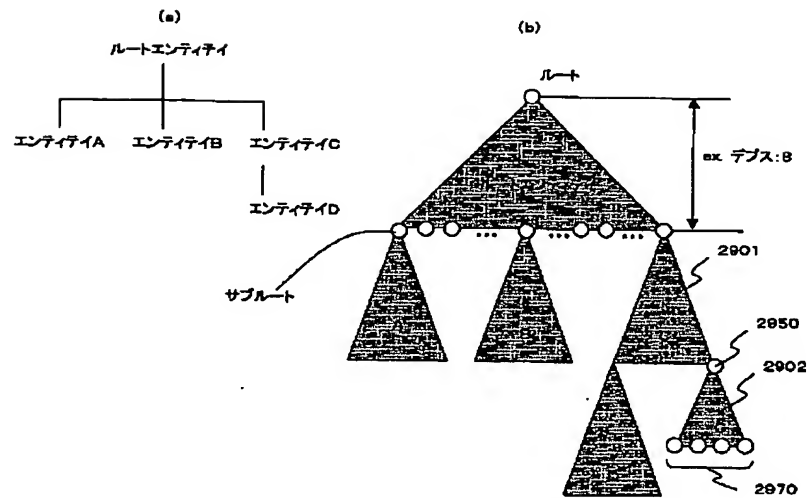
【図27】



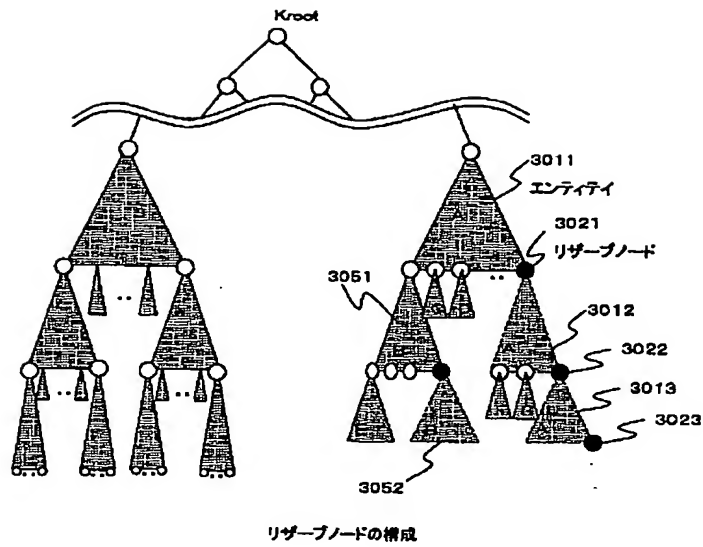
【図28】



【図29】

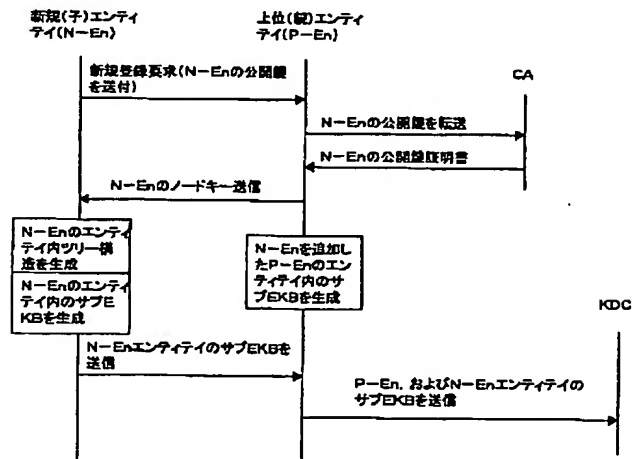


【図30】

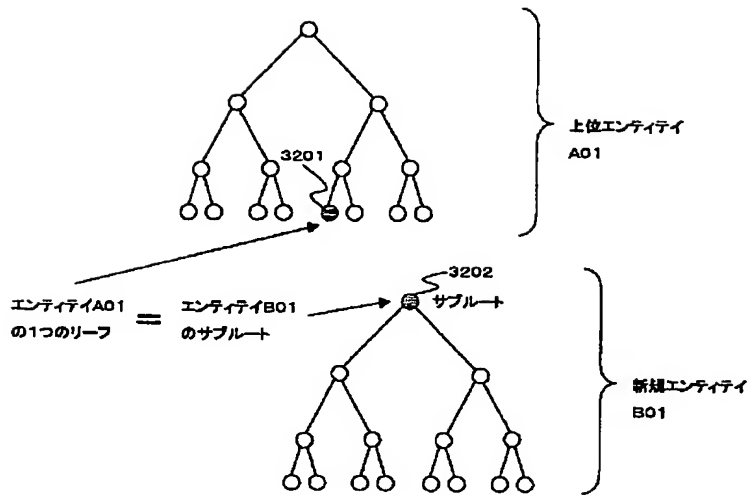


【図31】

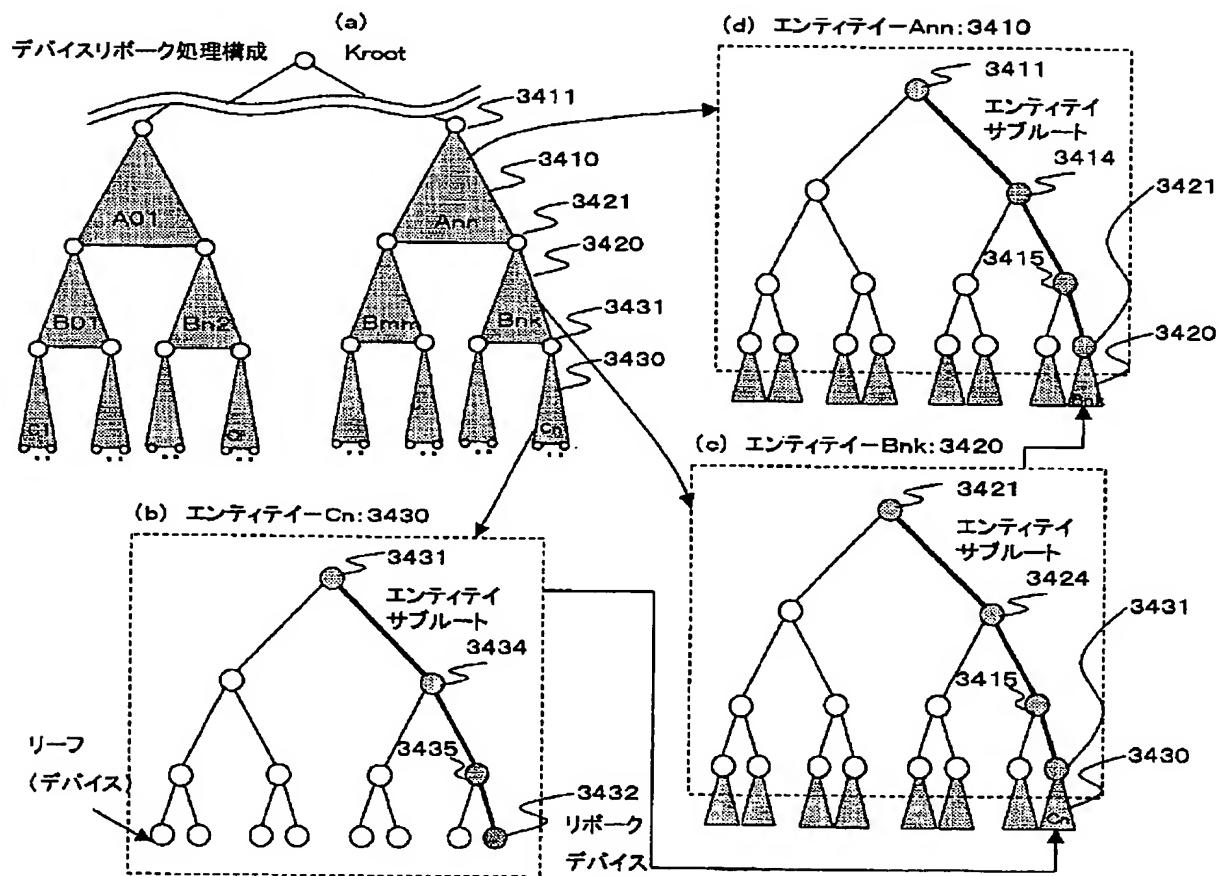
新規エンティティの登録処理



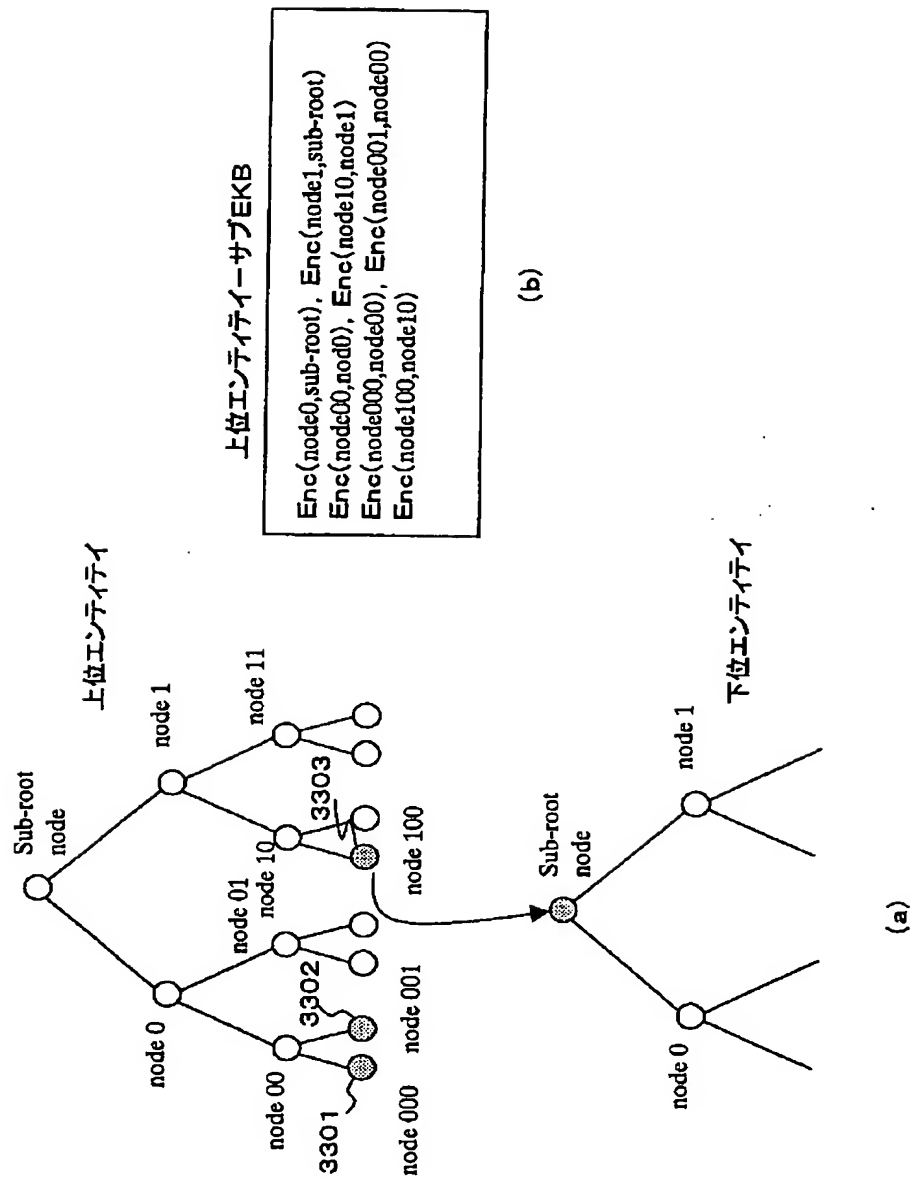
【図32】



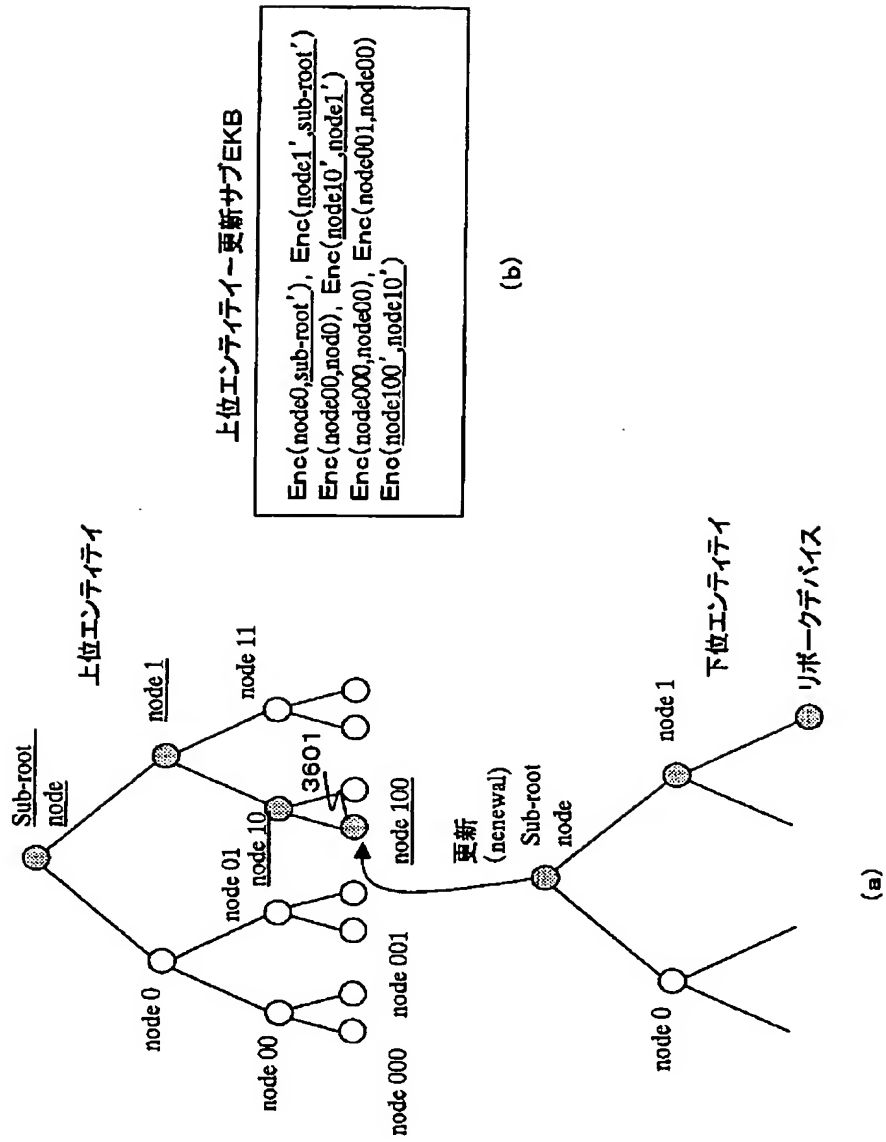
【図34】



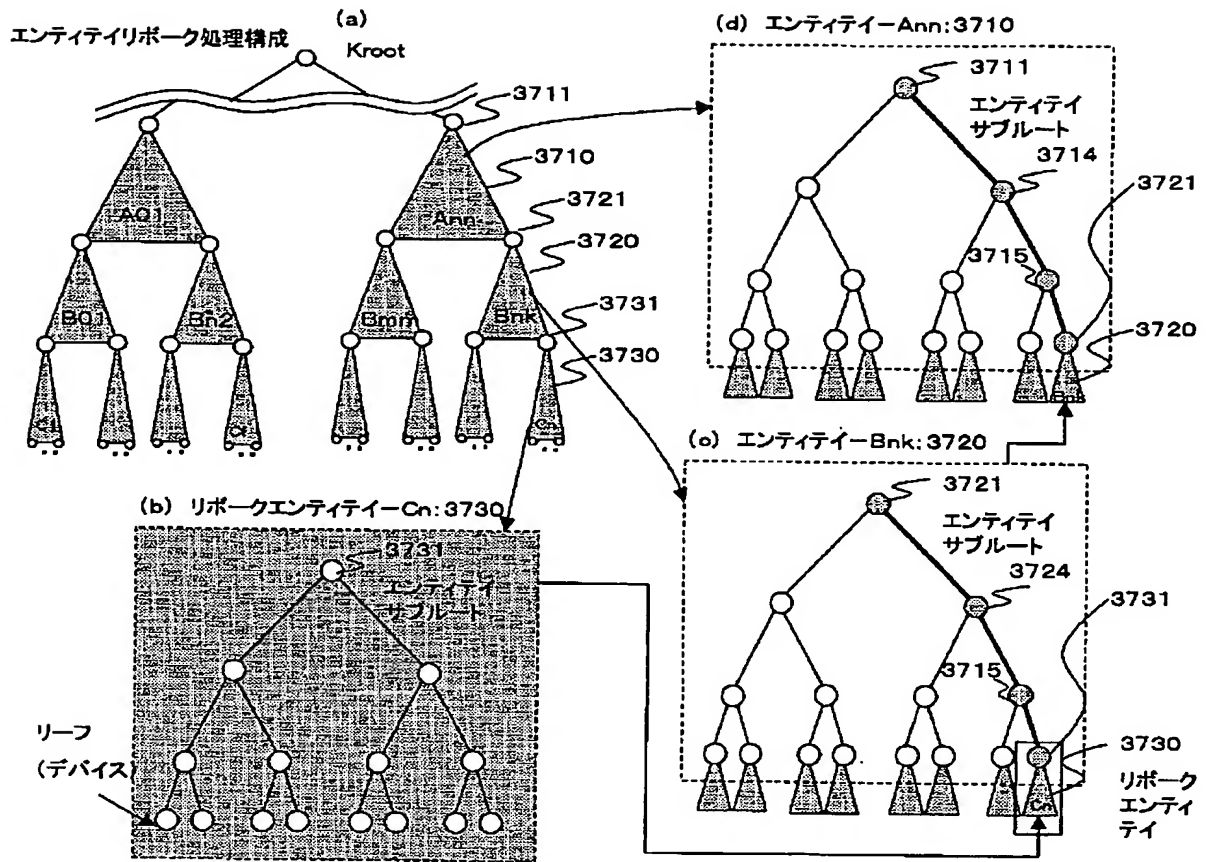
【图 3 3】



【図36】

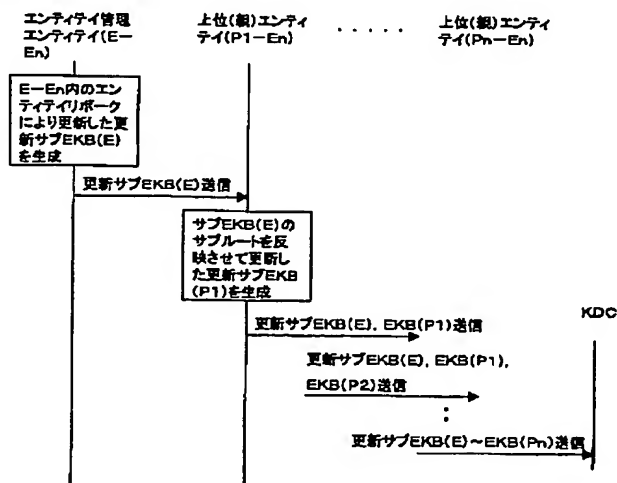


【図37】

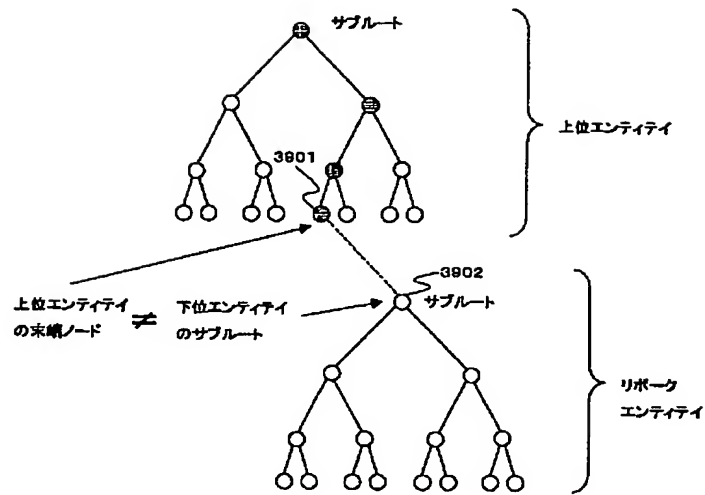


【図38】

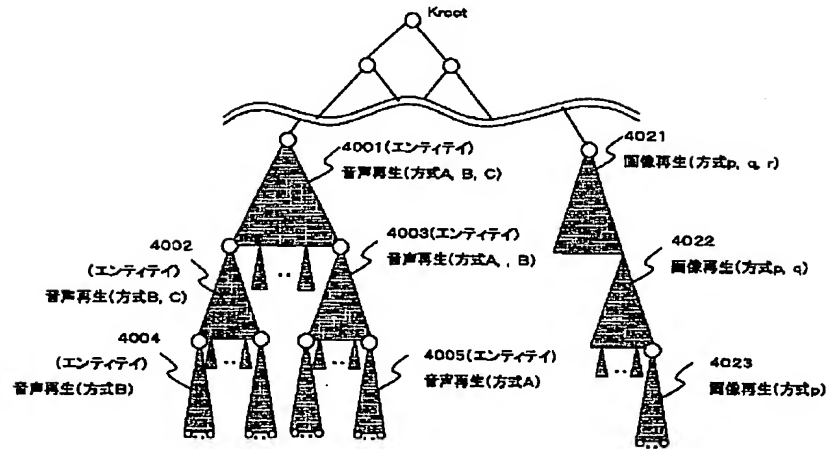
エンティティのリポーク処理



【図39】

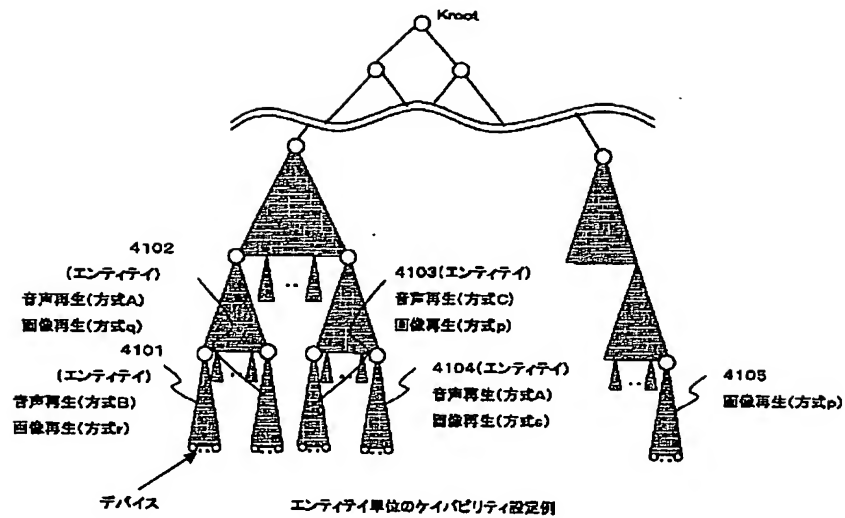


【図40】

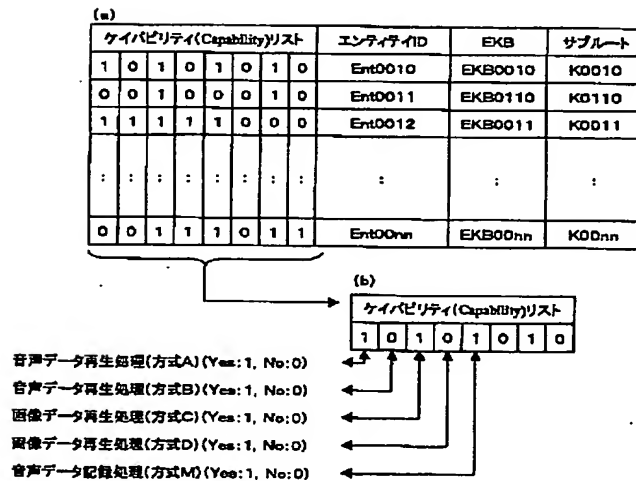


エンティティ単位のキャパシティ設定例

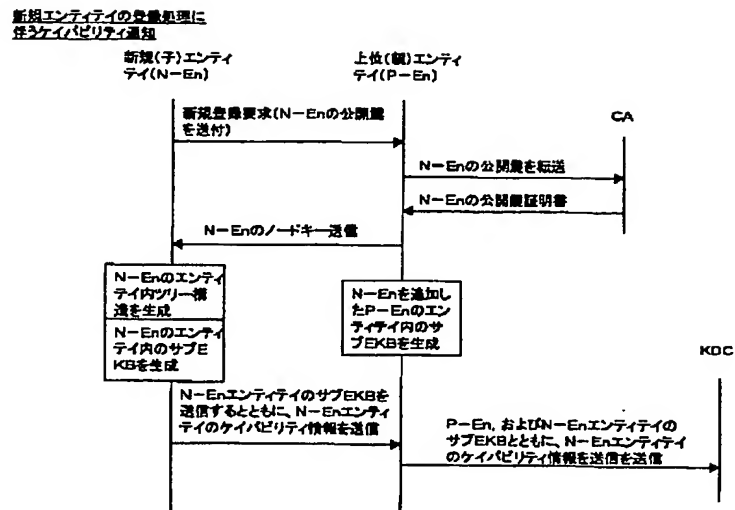
【図41】



【図42】



【図44】



フロントページの続き

| | | | |
|----------------------------|------|--------------|--------------------|
| (51) Int. Cl. ⁷ | 識別記号 | F I | テーマコード(参考) |
| H 0 4 L 9/08 | | H 0 4 L 9/00 | 6 0 1 B 6 0 1 A |

(72) 発明者 大澤 義知
 東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 浅野 智之
 東京都品川区北品川6丁目7番35号 ソニー株式会社内

F ターム(参考) 5B017 AA03 BA07 CA15 CA16
 5B082 EA11
 5B085 AA08 AE29
 5B089 GB03 JA33 JB22 KA08 KA17
 KC20 KH30
 5J104 AA01 AA16 EA02 EA04 EA06
 EA17 MA05 NA02 NA03 PA10

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☒ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☒ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☒ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

This Page Blank (uspto)